

UNIVERSIDAD CARLOS III DE MADRID

ESCUELA POLITÉCNICA SUPERIOR

DEPARTAMENTO DE INGENIERÍA TELEMÁTICA

PROYECTO FIN DE CARRERA

**Modelado fractal de tráfico de la red de Ingeniería
Telemática**

Autor:

Dña. Patricia Bravo García.

Departamento:

Ingeniería Telemática.

Tutor:

D. Carlos García Rubio.

Tribunal de Evaluación:

Presidente: D. Alberto García Martínez.

Vocal: Dña. Inmaculada Mora Jiménez

Secretario: D. Jesús Arias Fisteus.

Fecha de lectura: 24 de Junio de 2004

Calificación:



UNIVERSIDAD CARLOS III DE MADRID
ESCUELA POLITÉCNICA SUPERIOR

INGENIERÍA TÉCNICA DE TELECOMUNICACIÓN

PROYECTO FIN DE CARRERA

**Modelado fractal de tráfico de la red de Ingeniería
Telemática**

PATRICIA BRAVO GARCÍA

MADRID, 2004

Agradecimientos

Porque de buen nacido es ser agradecido, quiero dedicar este trabajo a mis padres, hermano y abuelos.

A mis profesores, especialmente a Luis Morán e Inmaculada Mora.

A David Blanco, Diana Martín, Soledad Hernández y Jessica Pérez.

A mi mejor amigo, Víctor Ruescas.

A todos, GRACIAS.

Resumen

El estudio de Leland en la década de los 90 estableció la base de considerar la auto-similitud como una noción importante en la comprensión del tráfico de red (incluyendo el modelado y el análisis de su rendimiento), y desde entonces se ha investigado mucho sobre la naturaleza de este fenómeno. Por ello, el tráfico de la red del departamento de Ingeniería Telemática presenta un reto de caracterización y modelado; así se dimensionarán correctamente los equipos y enlaces que conforman la red.

El objetivo es estudiar la naturaleza estadística del tráfico de este entorno LAN, presentar los principales desarrollos y resultados, y analizar sus implicaciones en el modelado de tráfico y la evaluación del rendimiento de red.

Las medidas de tráfico tomadas en el segmento de red de Ingeniería Telemática en Marzo de 2004, evidencian sus características impulsivas (*heavy-tailed*) en un amplio rango de escalas temporales, poniendo de manifiesto la propiedad de *long memory*.

La necesidad de este trabajo viene de la incapacidad de los modelos tradicionalmente considerados para caracterizar el tráfico telemático.

La transferencia de mensajes distribuidos según una variable aleatoria de cola pesada induce la auto-similitud en el tráfico de red; el grado con el que el tamaño de los ficheros es *heavy-tailed*, determina directamente el grado de auto-similitud del tráfico en el nivel de enlace.

Esta relación causal es robusta con respecto a cambios en los recursos de red (ancho de banda, capacidad de los buffers de los nodos), topología o cambios en la distribución del tiempo entre llegada de peticiones de ficheros.

Las propiedades del protocolo del nivel de transporte en la jerarquía TCP/IP juegan un rol importante para la preservación y modulación de esta relación.

La auto-similitud tiene algunas implicaciones negativas en el rendimiento de la red (en términos de tasa de pérdidas, tasa de retransmisiones, retardo en cola,...): el aumento de la tasa de pérdida de paquetes y del retardo en cola está directamente relacionado con el aumento del carácter auto-similar del tráfico.

Abstract

We demonstrate that Ethernet local area network (LAN) traffic is statistically self-similar, that none of the commonly used traffic models is able to capture this fractal behavior, and that such behavior has serious implications for the design, control, and analysis of high-speed, cell-based networks. Intuitively, the critical characteristic of this self-similar traffic is that there is no natural length of a "burst": at every time scale ranging from a few milliseconds to minutes and hours, similar-looking traffic bursts are evident; we find that aggregating streams of such traffic typically intensifies the self-similarity ("burstiness") instead of smoothing it. Our conclusions are supported by a rigorous statistical analysis of hundreds of millions of high quality Ethernet traffic measurements collected in 2004, coupled with a discussion of the underlying mathematical and statistical properties of self-similarity and their relationship with actual network behavior.

Índice general

| | |
|--|-----------|
| 1. Introducción | 17 |
| 1.1. Motivación | 17 |
| 1.2. Objetivos | 20 |
| 1.3. Contenido de la memoria | 20 |
| 1.4. Palabras clave | 21 |
| 2. Estado del arte | 23 |
| 2.1. Validación de un modelo | 24 |
| 2.2. Clasificación de los modelos | 25 |
| 2.2.1. Modelos de dependencias a corto plazo | 26 |
| 2.2.2. Modelos de dependencias a largo plazo | 27 |
| 2.3. Modelos incorrelados o ruidos blancos | 28 |
| 2.4. Modelo telefónico básico | 29 |
| 2.5. Modelo estadístico de los usuarios | 30 |
| 2.6. Modelos de sistemas | 30 |
| 2.7. Modelos del tráfico de datos en LANs | 32 |
| 2.8. Modelos clásicos de sistemas de colas | 32 |
| 2.9. <i>El fallo del modelo Poisson</i> | 32 |
| 2.10. Los tiempos entre llegadas no son exponenciales | 33 |
| 2.11. Ráfagas en todas las escalas | 33 |
| 2.12. ¿Siguiendo siendo válido un modelo de tráfico Poisson? | 34 |
| 2.13. Modelos tradicionales | 35 |
| 2.13.1. Modelos On-Off Poisson | 35 |
| 2.13.2. Modelos Markov | 36 |
| 2.13.3. Procesos modulados de Markov | 37 |
| 2.13.4. Modelos autoregresivos | 38 |
| 2.13.5. Modelos autoregresivos con media móvil | 38 |
| 2.13.6. Modelos autoregresivos integrados de media móvil | 39 |
| 2.14. Modelado fractal de tráfico | 39 |
| 3. Procesos auto-similares en redes de comunicaciones | 49 |
| 3.1. Algunos procesos auto-similares | 49 |
| 3.2. Procesos auto-similares continuos en el tiempo | 50 |

| | | |
|-----------|--|-----------|
| 3.3. | Procesos auto-similares discretos en el tiempo | 53 |
| 3.3.1. | Auto-similitud exacta de segundo orden | 54 |
| 3.3.2. | Auto-similitud asintótica de segundo orden | 56 |
| 3.4. | Propiedades del tráfico autosemejante | 57 |
| 3.5. | Dependencia a largo plazo y efecto Hurst | 57 |
| 3.6. | Lento decaimiento de las varianzas | 58 |
| 3.7. | Análisis estadístico de las medidas de tráfico Ethernet | 59 |
| 3.7.1. | Métodos estadísticos para comprobar la auto-similitud | 59 |
| 4. | Modelado de tráfico como multiplexación de conexiones de transporte | 65 |
| 4.1. | Estructura aditiva y gaussianidad | 65 |
| 4.1.1. | Auto-similitud a través de conexiones <i>heavy-tailed</i> | 66 |
| 4.2. | Variabilidad de la llegada de conexiones | 67 |
| 4.2.1. | Auto-similitud de la llegada de conexiones | 67 |
| 4.2.2. | Tiempo entre llegada de conexiones | 68 |
| 4.3. | Duración y tamaño de las conexiones | 68 |
| 4.4. | Tasa de transferencia de las conexiones | 70 |
| 4.5. | Efecto de la pila de protocolos | 72 |
| 4.6. | Implicaciones de la auto-similitud en el rendimiento | 73 |
| 4.7. | Modelado estocástico de la dependencia a largo plazo y las colas pesadas | 74 |
| 5. | Naturaleza fractal del tráfico IT UC3M | 77 |
| 5.1. | Medidas de tráfico | 77 |
| 5.2. | Evaluación de los estimadores sobre las trazas de tráfico | 91 |
| 5.2.1. | <i>Variance-time plot</i> | 91 |
| 5.2.2. | <i>R/S plot</i> | 92 |
| 5.2.3. | <i>Index dispersion counts</i> | 92 |
| 6. | Generación de tráfico fractal | 99 |
| 6.1. | Algoritmo FFT | 100 |
| 6.1.1. | Estimador Whittle | 100 |
| 6.1.2. | Método FFT (<i>Fast Fourier Transform</i>) | 101 |
| 6.2. | Algoritmo RMD | 104 |
| 6.3. | Modelo ON/OFF | 106 |
| 6.3.1. | Fuentes homogéneas | 107 |
| 6.3.2. | Comprobando el efecto "Noah" | 110 |
| 6.3.3. | Propiedad de robustez del efecto "Noah" | 111 |
| 6.3.4. | Naturaleza del tráfico generado por un usuario Ethernet individual | 112 |
| 6.3.5. | Implicaciones prácticas del efecto "Noah" | 113 |
| 6.3.6. | Generación y modelado de tráfico | 113 |
| 6.3.7. | Análisis de rendimiento | 114 |

| | |
|---|------------|
| 6.4. Resumen y trabajo futuro | 115 |
| 7. Modelado de tráfico web | 117 |
| 7.1. Background | 119 |
| 7.1.1. Protocolo HTTP | 119 |
| 7.1.2. Modelado de tráfico HTTP | 120 |
| 7.1.3. Colección del tráfico | 122 |
| 7.2. Modelo de tráfico | 122 |
| 7.2.1. ¿Por que los tiempos de transmisión se distribuyen según una variable aleatoria <i>heavy-tailed</i> ? | 125 |
| 7.2.2. La naturaleza del conjunto de ficheros únicos y la ac- ción de caché infinita | 127 |
| 7.2.3. ¿Por qué los ficheros disponibles en el web son <i>heavy- tailed</i> ? | 129 |
| 7.2.4. Implicaciones para el tráfico | 133 |
| 8. Modelado de tráfico DHCP | 135 |
| 8.1. Un primer intento de automatizar la configuración de dispos- itivos | 135 |
| 8.2. Asignando direcciones IP utilizando DHCP | 136 |
| 8.2.1. Regeneración de direcciones con DHCP | 136 |
| 8.2.2. Describiendo servicios de red con DHCP | 136 |
| 8.2.3. Añadiendo servicios con DHCP | 137 |
| 8.3. Problemas del protocolo DHCP | 137 |
| 8.3.1. Carga del servidor | 137 |
| 8.4. Estados del cliente | 137 |
| 8.4.1. Obteniendo una configuración inicial | 138 |
| 9. Conclusiones y líneas futuras | 147 |
| 9.1. Conclusiones | 147 |
| 9.2. Líneas futuras | 148 |
| 9.2.1. Hacia modelos de carga de trabajo auto-similares no gaussianos | 148 |
| 9.2.2. El comportamiento del tráfico de red en pequeñas es- calas de tiempo | 150 |
| A. Selfis | 151 |
| B. Analizadores de protocolos | 153 |
| B.1. Ethereal | 154 |
| B.2. TCPDump | 154 |
| B.2.1. Interpretando la salida | 156 |
| C. Tcp-Reduce | 159 |

| | |
|--|------------|
| D. Presupuesto del proyecto | 163 |
| D.1. Presupuesto de ejecución material | 163 |
| D.1.1. Gastos de material | 163 |
| D.2. Coste de mano de obra | 164 |
| D.2.1. Dirección del proyecto | 164 |
| D.2.2. Importe total del proyecto | 164 |

Índice de figuras

| | |
|---|-----|
| 2.1. Tráfico real (izquierda) | 40 |
| 4.1. Influencia del nivel de aplicación en la auto-similitud del nivel de enlace. | 72 |
| 5.1. Escenario de la red de Ingeniería Telemática | 77 |
| 5.2. Jerarquía de protocolos TCP/IP | 85 |
| 5.3. cdf (<i>cumulative distribution function</i>) y ccdf (<i>complementary cumulative distribution function</i>) del tiempo entre conexiones TCP. | 87 |
| 5.4. cdf (<i>cumulative distribution function</i>) y ccdf (<i>complementary cumulative distribution function</i>) de la duración de las conexiones TCP. | 88 |
| 5.5. cdf (<i>cumulative distribution function</i>) y ccdf (<i>complementary cumulative distribution function</i>) de la tasa de transferencia en sentido ascendente. | 89 |
| 5.6. cdf (<i>cumulative distribution function</i>) y ccdf (<i>complementary cumulative distribution function</i>) de la tasa de transferencia en sentido descendente. | 90 |
| 5.7. Nivel de agregación 100 segundos | 93 |
| 5.8. Nivel de agregación 10 segundos | 94 |
| 5.9. Nivel de agregación 1 segundo | 95 |
| 5.10. Nivel de agregación 0.1 segundos | 96 |
| 5.11. Nivel de agregación 0.01 segundos | 97 |
| 6.1. Ilustración de las tres primeras iteraciones del algoritmo RMD | 105 |
| 6.2. Relación entre $Z((a+b)/2)$, $Z(a)$, $Z(b)$ y Z_{RMD} | 105 |
| 6.3. <i>Renewal reward process</i> | 107 |
| 7.1. Clientes, red y servidores en el world wide web | 118 |
| 7.2. Implementación de las conexiones cliente al web | 119 |
| 7.3. cdf (<i>cumulative distribution function</i>) de la duración de las conexiones www. | 124 |

| | | |
|------|--|-----|
| 7.4. | ccdf (<i>complementary cumulative distribution function</i>) de la duración de las conexiones <i>www</i> | 125 |
| 7.5. | cdf (<i>cumulative distribution function</i>) y ccdf (<i>complementary cumulative distribution function</i>) del número de bytes transmitidos en las conexiones <i>www</i> | 126 |
| 7.6. | Relación entre los conjuntos de datos | 129 |
| 7.7. | CD del tamaño de los ficheros de 32 sitios web | 130 |
| 7.8. | cdf (<i>cumulative distribution function</i>) y ccdf (<i>complementary cumulative distribution function</i>) de la tasa de transferencia de las conexiones <i>www</i> (<i>upstream</i>). | 131 |
| 7.9. | cdf (<i>cumulative distribution function</i>) y ccdf (<i>complementary cumulative distribution function</i>) de la tasa de transferencia de las conexiones <i>www</i> (<i>downstream</i>). | 132 |
| 8.1. | Una máquina de estados finitos para un cliente DHCP | 138 |
| 8.2. | cdf (<i>cumulative distribution function</i>) y ccdf (<i>complementary cumulative distribution function</i>) del tiempo transcurrido entre mensajes <i>Request</i> | 143 |
| 8.3. | cdf (<i>cumulative distribution function</i>) y ccdf (<i>complementary cumulative distribution function</i>) del tiempo transcurrido entre mensajes <i>Decline</i> | 144 |
| 8.4. | cdf (<i>cumulative distribution function</i>) y ccdf (<i>complementary cumulative distribution function</i>) del tiempo transcurrido entre mensajes <i>Offer</i> | 145 |

Índice de cuadros

| | |
|---|-----|
| 5.1. Estadísticos de la tasa de paquetes transmitidos en distintas escalas de tiempo. | 79 |
| 5.2. Estadísticos de la tasa de paquetes transmitidos en distintas escalas de tiempo. | 82 |
| 5.3. Estadísticos generales para las conexiones TCP | 83 |
| 5.4. Bytes (<i>DS</i> y <i>US</i>) transmitidos durante el periodo de monitorización | 84 |
| 5.5. Conexiones TCP establecidas en el periodo de monitorización | 84 |
| 5.6. Protocolos monitorizados | 85 |
| 5.7. Valores de la pendiente de la recta que aproxima (mediante la técnica de mínimos cuadrados) la nube de puntos de la estimación | 98 |
| 5.8. Valores estimados de H (\hat{H}) | 98 |
| 7.1. Estadísticos de las conexiones www. | 133 |
| 8.1. Mensajes DHCP | 142 |
| D.1. Gastos de material | 163 |
| D.2. Costes mano de obra | 164 |
| D.3. Coste total del proyecto | 164 |

Capítulo 1

Introducción

1.1. Motivación

La geometría fractal apareció en la década de los 70 como una síntesis de los trabajos desarrollados principalmente en las matemáticas y en la física durante más de un siglo. Esta nueva geometría de objetos irregulares constituye la base de un profundo conocimiento de ciertos dominios y también ha permitido el diseño de mejores modelos para los análisis financieros y algoritmos más robustos para el procesamiento de imágenes.

Una definición básica de **fractal** es la siguiente: objeto cuyo comportamiento no varía a pesar del cambio de escala - no importa cuánto se aproxime el zoon en una parte de la señal porque se va a observar un comportamiento similar; gran parte de los trabajos matemáticos aplicados en este área se encargan de dar un significado preciso a esto.

La idea subyacente es bastante simple: **investigar el comportamiento del fenómeno estudiado a distintas escalas**. En vez de fijar una resolución para el análisis, se considera simultáneamente un amplio rango de resoluciones y de esto se infiere una información útil. El presente trabajo examina cómo esta idea tiene una importancia significativa en el análisis de tráfico de datos de la red del departamento de Ingeniería Telemática.¹

Aunque el tráfico es puramente *man - made*, debe ser estudiado como un fenómeno natural (muchos de sus aspectos van más allá del alcance de este trabajo). En otras palabras, mientras que los componentes "atómicos" del tráfico son diseñados y controlados, su interacción resulta en comportamientos nuevos (cualitativa y cuantitativamente) que deben ser descubiertos y analizados.

Las causas de la complejidad del tráfico se clasifican en cuatro categorías

1. La estructura multicapa de la red - enlaces de subredes y nodos, cada uno con su propia complejidad y diversidad en términos de conectivi-

¹Siguiendo una histórica perspectiva, la atención se centrará en la dependencia a largo plazo (*long range dependence*)

dad, ancho de banda e implementación física. Ésto favorece la emergencia simultánea de comportamientos distintos.

2. La segunda categoría está directamente relacionada con el flujo de datos:
 - a) Dinámica asociada a los protocolos multicapa (mecanismos y escalas de tiempo).
 - b) Segmentación y combinación de los datos; los flujos de paquetes se corresponden con una mezcla compleja de fuentes.
3. Variedad (en sus características y demanda) de aplicaciones ofrecidas como servicios de la red.
4. Variabilidad del tráfico en un amplio rango de escalas de tiempo.

Todas las escalas temporales son importantes para un aspecto del tráfico u otro

- Microsegundos: escala de tiempo característica de las LANs - es la duración necesaria para transferir un Megabyte en una LAN de capacidad 1 Gigabit (subred del departamento de Ingeniería Telemática).
- Milisegundos: escala de tiempo para las colas en los gateways grandes y también para los mecanismos de control de admisión y control de congestión.

Es probable que, con el progreso de la tecnología, continúe creciendo el número de escalas de interés. Es importante comprender las relaciones entre los fenómenos que aparecen en las diferentes escalas de tiempo; este es precisamente el objetivo del análisis fractal. El tráfico LAN no se caracteriza únicamente por la existencia de múltiples escalas de tiempo de interés, sino también por la fuerte relación entre ellas.

Viendo el tráfico de datos como un fenómeno natural, parece que las soluciones *man-made* para el desarrollo de las redes LAN, responde a situaciones concretas. Parece natural esperar que un cuidadoso análisis fractal del tráfico de la red del departamento de Ingeniería Telemática revele características cruciales que puedan ayudar a mejorar su rendimiento.

El modelo más simple para el tráfico telefónico clásica emplea procesos Poisson. Una de las propiedades más importantes de los procesos Poisson es su "falta de memoria" (*memoryless*); los eventos futuros son básicamente independientes de los eventos pasados.

Basta observar las trazas del tráfico Ethernet de la red de Ingeniería Telemática para comprobar el contraste entre el tráfico de datos y el tráfico de voz; la variabilidad (*burstiness*) del primero implica que los eventos ocurren de forma agrupada y por tanto pueden estar correlados.

Al principio, los ingenieros de tráfico pensaban que esta complejidad añadida podía ser solucionada considerando procesos estocásticos que fueran más complejos que los procesos Poisson, pero sólo se obtuvieron correlaciones a corto plazo, no independencia; en la década de 1990 se aceptó la necesidad de otro modelo.

Considerando los argumentos de escala descritos anteriormente, se puede garantizar que las ráfagas de tráfico no desaparecen después de realizar un promedio temporal; en ninguna resolución la representación gráfica se aplana, siempre se observa el mismo comportamiento

Desde un punto de visto estadístico, ésto significa que los procesos válidos como modelos relevantes son aquellos que presentan correlaciones en todas las escalas de tiempo. Estos procesos son conocidos en la literatura matemática como procesos LRD (*long range dependence*) y se caracterizan por una función de autocorrelación que decae como k^β footnote La función de autocorrelación mide el "grado de dependencia" entre lo que ocurre en el instante de tiempo t y $t + k$; el exponente β se denomina exponente LRD y puede tomar cualquier valor del intervalo $[0, 1]$.

La forma de la función de autocorrelación implica que la autocorrelación total (suma en k de todas las correlaciones) tiende a infinito. Ésto da un significado preciso a la intuición de las fuertes correlaciones involucradas en la variabilidad en todas las escalas de tiempo.

El decaimiento hiperbólico de la función de autocorrelación se ha observado en casi todas las áreas del tráfico LAN (tamaño de ficheros, tiempos de transmisión, ...).

El modelo más ampliamente aceptado que explica la emergencia del fenómeno LRD se basa en la superposición de fuentes *onoff*. En particular, una secuencia de fuentes Poisson con un decaimiento hiperbólico (variable aleatoria heavy-tailed) para los periodos de actividad e inactividad es suficiente para generar LRD.

LRD tiene repercusiones extremadamente importantes en el rendimiento de la red. En presencia de este fenómeno, el comportamiento de la cola del sistema es completamente diferente del comportamiento clásico; se puede observar un decrecimiento potencial para el tamaño de la cola, llevando, en algunas situaciones, a un tiempo de espera infinito.

LRD se refiere al comportamiento del tráfico en frecuencias bajas, ya que se está examinando el decaimiento de las correlaciones en escalas de tiempo cada vez más grandes. Sin embargo, en las trazas de tráfico la variabilidad (*burstiness*) es observada desde escalas de tiempo pequeñas. Investigar el significado y las propiedades del comportamiento del tráfico en altas frecuencias resulta mucho más complejo y no se va a abordar en este trabajo.

1.2. Objetivos

El objetivo de este documento es establecer de una manera estadísticamente rigurosa la característica de la auto-similitud o naturaleza fractal de las medidas de tráfico Ethernet de alta resolución temporal coleccionadas en la red del departamento de Ingeniería Telemática. Detalladamente, los objetivos son:

- Revisar las características de los modelos de tráfico de paquetes tradicionalmente empleados en la literatura.
- Justificar la no validez de estos modelos para representar el tráfico de la red de Ingeniería Telemática.
- Ilustrar algunas de las diferencias más llamativas entre los modelos tradicionales y los modelos auto-similares (o fractales).
- Presentar las propiedades de los modelos auto-similares.
- Estimar el parámetro de Hurst para conocer el carácter auto-similar del tráfico de la red de Ingeniería Telemática. También se van a presentar otros análisis para completar el estudio estadístico.
- Explicar la naturaleza auto-similar del tráfico a partir de la multiplexación de conexiones TCP.
- Analizar la influencia de los protocolos de nivel de transporte en la propagación de la auto-similitud desde el nivel de aplicación de la arquitectura TCP/IP al nivel de enlace.
- Presentar algunas de las implicaciones del tráfico auto-similar de red en el diseño, control y análisis de rendimiento de la red.
- Presentar dos algoritmos de generación sintética de tráfico fractal basados en procesos auto-similares: RMD y FFT.
- Estudiar en detalle el tráfico web y el tráfico dhcp.

1.3. Contenido de la memoria

Los capítulos de este proyecto cubren medidas de tráfico, modelado, análisis de rendimiento y control de tráfico para tráfico de red auto-similar.

En el primer capítulo se presentan los modelos de teletráfico considerados en la literatura hasta 1993, conocidos como "modelos clásicos". Se analizan los motivos que justifican la transición hacia el uso de los modelos fractales o auto-similares.

En el segundo capítulo se presentan los fundamentos matemáticos y las propiedades de los procesos estocásticos auto-similares. Asimismo se estudian detalladamente los distintos grados de auto-similitud: exacta, asintótica y estricta.

En el tercer bloque de este trabajo se estudian las causas del carácter autosemejante del tráfico de la red de Ingeniería Telemática. Se analiza la distribución estadística de las variables que intervienen en el modelado de tráfico y a partir de este punto se deriva la auto-similitud del tráfico conjunto.

En el cuarto capítulo se realiza una descripción estadística de las trazas de tráfico coleccionadas en la red de Ingeniería Telemática en Marzo 2004. Se va a estimar el valor del parámetro de Hurst (que define el carácter auto-similar) del tráfico empleando tres técnicas diferentes.

En el quinto capítulo se va a realizar una discusión sobre dos diferentes aproximaciones para modelar el tráfico de red auto-similar: RMD, FFT. En este bloque, también, se va a explicar en detalle el modelo *on/off* de varianza infinita.

En los capítulos seis y siete se caracteriza con un alto nivel de detalle el tráfico generado por los servicios *web* y *dhcp*.

En último lugar se presentan algunas conclusiones y futuras líneas de trabajo.

1.4. Palabras clave

Auto-similitud o autosemejanza (SS), efecto Hurst, lento decaimiento de la varianza, dependencia a largo plazo (LRD), dependencia a corto plazo (SRD), variabilidad, distribución de cola pesada (*heavy-tailed*), IDC, FGN (*Fractional Gaussian Noise*), FARIMA (*Fractional Autoregressive Moving Average*), FFT (*Fast Fourier Transform*), RMD (*Random Midpoint Displacement*)).

Capítulo 2

Estado del arte

La necesidad de conocer el comportamiento del tráfico telemático subyace a toda discusión relacionada con el dimensionamiento y diseño de controles en redes de telecomunicación; el modelado analítico de las fuentes de tráfico en una red es la base de su diseño.

El objetivo de reemplazar el tráfico medido por modelos matemáticos es obtener un rendimiento similar al que se tiene en el entorno real. Así, las medidas de tráfico a gran escala pueden ser reducidas a la evaluación de unos pocos parámetros, combinados con alguna indicación de su evolución a lo largo del tiempo; la simplicidad del modelo facilita la evaluación analítica del rendimiento de la red (o de algunos de sus elementos).

En definitiva, un modelo de tráfico es una abstracción matemática más o menos compleja que trata de imitar las propiedades estadísticas del tráfico real. La primera fase del diseño es un proceso de **caracterización**, que consiste en el análisis de la realidad a imitar (tráfico de red). El objetivo es describirla mediante un reducido conjunto de variables (descriptor estadístico), que resuma y abstraiga sus aspectos más importantes; especificar la distribución marginal y la función de autocorrelación de un proceso no es suficiente para garantizar la exactitud en la predicción de los resultados de rendimiento.

Tras la definición de este descriptor, la atención del modelado se centra en elegir un proceso matemático que lo aproxime (con independencia de la realidad física que representa); la secuencia de valores generada a partir de este proceso tendrá propiedades similares a las del flujo de datos observado.

Un aspecto clave a la hora de definir un modelo es la escala de tiempos a la que el modelo debe representar convenientemente la realidad. A menudo resulta costoso modelar las fluctuaciones que observadas en un amplio rango de escalas temporales.

La correspondencia con la realidad no es el único baremo que se debe considerar en la elección de una estrategia de modelado. Así, resultan de interés otros aspectos:

- Tratabilidad analítica. El modelo debe arrojar resultados analíticos sin

necesidad de simulación.

- Facilidad de implementación (vía software o hardware). La estructura, la algoritmia y los costes computacionales que exige la implementación del modelo deben ser simples.
- Parsimonia. El modelo debe ser caracterizado con un número reducido de parámetros (con un significado físico asociado).
- Capacidad de modelar otros tipos de tráfico y, en especial, tráfico agregado. En numerosas ocasiones se puede exigir a un modelo la ductilidad suficiente para que, mediante ciertos cambios de sus parámetros, éste sea capaz de imitar otros tipos de tráfico o incluso el propio tráfico agregado que resulta de multiplexar sobre un canal varias fuentes individuales.

2.1. Validación de un modelo

¿En que consiste la imitación que hace el modelo de la realidad? ¿Cómo debe ser evaluada esta imitación? Algunas de las comparaciones que se pueden establecer entre el comportamiento real del tráfico y el comportamiento simulado, son las siguientes:

- Comparar visualmente la evolución en el tiempo de la muestra real con la generada por una ejecución del modelo propuesto.
Esta evaluación constituye una validación meramente cualitativa, pero es un método adecuado para localizar características de la señal de una manera rápida. La validación visual no basta para aceptar un modelo pero sí sirve para rechazarlo.
- Estimar ciertos estadísticos (de primer y segundo orden) y compararlos con los del tráfico real.
- Analizar el comportamiento del sistema suponiendo como entrada la señal real y una ejecución del modelo propuesto, y cotejar los resultados de ambos análisis.

Esta validación es más realista que las anteriores y consiste en:

1. Inyectar tráfico real en el sistema.
 2. Estimar una serie de estadísticos (tasa de pérdidas o retrasos medios).
 3. Comparar estos resultados con los obtenidos a partir del modelo.
- En última instancia la validez de un modelo viene determinada por su capacidad de imitar el tráfico en el entorno concreto donde se pretende aplicar.

Los esquemas anteriores no dejan de ser "tests intermedios", de los que no se puede inferir nada definitivo acerca de la validez del modelo. Así, el modelo deberá ser evaluado (determinando el error que se comete) con relación a la realidad.

2.2. Clasificación de los modelos

La clasificación de los modelos de teletráfico se basa en el análisis de la función de autocorrelación del proceso estocástico subyacente. La función de autocorrelación de cierta señal es un indicador del grado de dependencia lineal que existe entre una muestra y la k-ésima muestra anterior:

$$R_X(k) = \frac{1}{\sigma_X^2} E[(X[i+k] - \mu_X)(X[i] - \mu_X)]$$

donde μ_X representa la media de la señal $X[n]$ y σ^2 su varianza. En el modelado de tráfico telemático la señal $X[n]$ lleva aparejada la evolución temporal de una variable física (número de usuarios conectados, duración de una llamada, número de bytes transmitidos en algún intervalo de tiempo).

De acuerdo con la forma que el modelo impone a esta función, se pueden distinguir:

1. Modelos dependientes a corto plazo (*short range dependent models*).
2. Modelos dependientes a largo plazo (*long range dependent models*).¹

¹Ambos son estacionarios.

2.2.1. Modelos de dependencias a corto plazo

El carácter a ráfagas del tráfico telemático impuso la necesidad de introducir correlación en los modelos que lo representaban. Las primeras soluciones pasaron por cadenas markovianas de pocos estados (la correlación viene dada por la dependencia del estado actual con el estado anterior). Los procesos modulados por Markov, especialmente el proceso MMPP (*Markov Modulated Poisson Process*) de dos estados, resultan apropiados para modelar el agregado de fuentes *on-off* (la generación del tráfico evoluciona entre dos estados con generación poissoniana de distinta tasa media). Una ventaja de los modelos markovianos es su tratabilidad analítica en teoría de colas, pero la necesidad de definir una matriz con las probabilidades de conmutación entre estados, motiva un crecimiento geométrico del número de parámetros conforme aumenta el número de estados. En este caso, y a costa de perder la mencionada tratabilidad, se puede recurrir a los filtros ARMA (*Auto regressive moving average*), en los que el valor de la muestra actual viene dado por los N valores anteriores y M valores de un ruido blanco aleatorio denominado proceso de innovación. En cualquier caso, los procesos markovianos, los filtros ARMA, y alguna otra solución más compleja que pudiera ser incluida en este grupo, comparten algunas propiedades:

1. Carecen de la suficiente variabilidad para representar el tráfico de datos.
2. La representación en grandes escalas de tiempo equivale a ruido blanco gaussiano.
3. No son capaces de aproximar la función de autocorrelación de la señal de tráfico para valores altos del retardo k.

Estos procesos se diseñan para ajustar las dependencias a corto plazo o SRD (*short range dependence*) porque poseen una función de correlación que decae exponencialmente.

$$R_X(k) \sim e^{-k} \quad k \rightarrow \infty$$

2.2.2. Modelos de dependencias a largo plazo

Al principio de la década de los 90, momento en que Leland y otros autores publicaron un trabajo en el que se probaba la naturaleza auto-semejante de muestras de tráfico Ethernet [Leland et al., 1993], se comenzó a estudiar la presencia de un fenómeno estadístico en el tráfico de redes telemáticas: **dependencia a largo plazo o LRD (*long range dependence*)**.

Este fenómeno se refleja en coeficientes de correlación con decaimiento hiperbólico y se asocia a la existencia de una variabilidad del tráfico en una amplia gama de escalas temporales; no modelar esta variabilidad acarrea consecuencias graves en muchos esquemas de control, que asumen el carácter intermitente del tráfico para una escala "natural" y acotada.

La **variabilidad** de la señal se liga a cierto grado de **autosemejanza o fractalidad**, que ha permitido aplicar en este campo las conclusiones y la analítica desarrollada por Mandelbrot [B.B.Mandelbrot,1996] y otros autores que ya habían estudiado procesos fractales en otras ramas de la ciencia.

El fenómeno de dependencia a largo plazo en series temporales ha venido a revolucionar, al menos parcialmente, el ámbito del modelado de tráfico telemático, ya que se ha detectado en el tráfico IP.

El argumento esgrimido [A. Erramilli et al., 1996] para explicar este fenómeno en series temporales, es la existencia de una estructura multi-nivel de generación. Así, para justificar la LRD del tráfico web se apela a la interacción del comportamiento del usuario (preferencias por ciertos ficheros, tiempo de análisis de páginas web, ...) con las políticas de caché de los buscadores, el tamaño subexponencial o hiperbólico de los objetos en las páginas web, y el agregado de tráfico que resulta en la red de área local.

Hasta la fecha existen dos técnicas genéricas en las que se podrían agrupar los modelos planteados para aproximar la naturaleza LRD de una señal de tráfico telemático:

Modelos multinivel: imitan directamente la variabilidad existente en algunos niveles o escalas de tiempo. Estos modelos "quasi-fractales" presentan estructuras complicadas, de escasa tratabilidad y su definición requiere un número amplio de parámetros, lo que perjudica la parsimonia del modelo.

Modelos autosemejantes: estos modelos ya responden a estructuras estrictamente fractales, es decir, de variabilidad en todas las escalas de tiempo. Se corresponden con modelos bien conocidos, procesos FGN (*Fractional Gaussian Noise*) o procesos F-ARIMA.

Con los modelos del primer grupo se pueden generar señales fractales puras (se mantiene la misma distribución estadística en todas las escalas de tiempo), y los modelos FARIMA permiten añadir el ajuste de las dependencias a corto plazo incorporando una típica estructura autorregresiva de media móvil (ARMA).

La principal ventaja de los modelos autosemejantes frente a los modelos multinivel, es su capacidad de representar el fenómeno LRD (*long range dependence*) mediante un único parámetro, parámetro de Hurst o parámetro H, que denota el grado de autosemejanza de la muestra a modelar.

A pesar de la parsimonia, estimar el valor de H de una forma ajustada e insesgada resulta muy difícil por la propia naturaleza de los estimadores propuestos. Dichos estimadores se basan en el análisis de comportamientos asintóticos (*rescaled adjusted range, variance time plot*) o en la minimización de funciones que suponen previamente una forma al espectro (*periodogram*), e introducen siempre cierto sesgo en la medida, de por sí limitada a un estrecho intervalo [0.5 , - 1]. Esta problemática se agrava cuando los estimadores se aplican sobre series de tráfico real finitas y con posibles problemas de no estacionariedad.

Así pues, la aproximación a la naturaleza LRD exige decidir en un compromiso a favor de la parsimonia o de la complejidad de medida y generación.

Antes de estudiar en detalle los modelos autosemejantes, se van a revisar los principios de los modelos tradicionalmente empleados en el área de teletráfico (en redes conmutadas de circuitos y paquetes).

2.3. Modelos incorrelados o ruidos blancos

Se trata de señales donde no existe relación de ningún tipo entre las muestras de la secuencia

$$R_X(k) = 0 \quad \forall k \neq 0$$

La generación de cada muestra es independiente pero todas ellas siguen la misma distribución estadística.

En los últimos años han cobrado una importancia crucial los procesos con varianza infinita o de cola hiperbólica o pesada (*heavy-tailed*), pero hasta entonces, los modelos estadísticamente independientes (i.i.d *independent identically distributed*) se consideraban válidos para modelar ciertos aspectos de la generación del tráfico (tamaño de los ficheros transmitidos, duración de las llamadas, ...).

El modelo Poisson y los modelos de ruido gaussiano se clasifican dentro de este grupo. De especial interés resultan los procesos distribuidos con la función logaritmo-normal, por cuanto se ha demostrado que la percepción del tiempo por parte del usuario de servicios de telecomunicación sigue una distribución normal en escala logarítmica.

2.4. Modelo telefónico básico

En la actualidad, uno de los servicios de telecomunicaciones más difundido es la telefonía básica. Desde sus comienzos (hace más de un siglo) hasta nuestros días, se ha convertido en un servicio de comunicaciones con gran importancia a nivel de usuario residencial, empresarial, o incluso a nivel estratégico y táctico gubernamental.

Este servicio presenta al usuario final una calidad de servicio determinista, una vez establecida la llamada con el otro extremo la comunicación goza de un canal reservado de ancho de banda constante (la voz no sufre cortes y la variación del retardo está limitada). La reserva de un circuito con un ancho de banda fijo se basa en el análisis de la señal que va a circular por este canal, la señal de voz.

Durante el funcionamiento normal de la red la mayor parte del equipamiento no se emplea, por tanto la topología de la red telefónica es jerárquica, no existe un enlace por cada par de abonados. La comunicación entre abonados pertenecientes a diferentes centrales finales se realiza estableciendo un circuito que atraviesa otras centrales de tránsito en niveles superiores de la jerarquía.

Los enlaces entre las centrales finales y los nodos de tránsito soportan un gran número de circuitos simultáneos gracias a técnicas de multiplexación en frecuencia o en tiempo. El objetivo de dimensionamiento en estas redes es determinar la capacidad (en número de circuitos de voz) de estos enlaces que permita dar la mayor disponibilidad al menor coste. Los problemas se centran en torno a dos situaciones en las cuales es imposible establecer el circuito deseado entre dos abonados y por tanto se pierde una llamada por falta de disponibilidad.

1. Bloqueo interno de algunos conmutadores en el trayecto del circuito, es decir, imposibilidad de establecer ese circuito debido a falta de recursos internos del conmutador. Esta probabilidad de bloqueo depende de la arquitectura escogida para el conmutador.
2. Saturación de todos los canales existentes entre nodos finales y de tránsito o entre estos últimos en el trayecto del circuito.

Para predecir el comportamiento de la red ante la llegada de usuarios, es necesario una descripción estadística de su comportamiento respecto a los

instantes en los que solicitan servicio, la duración del mismo y la distancia con el destino (bien sean llamadas locales a la central o no).

2.5. Modelo estadístico de los usuarios

La atención se va a centrar en el modelado del comportamiento de un conjunto de usuarios, no de los usuarios de forma individual. Las variables de interés son el patrón de generación de llamadas y su duración.

La secuencia de valores correspondientes al tiempo entre llamadas en una población de usuarios, se ajusta correctamente a un proceso estocástico $T = (T_k : k = 0, 1, 2, \dots)$ semi-infinito de tiempo discreto y estado continuo donde los T_k son variables aleatorias i.i.d.s (independientes e idénticamente distribuidas) que siguen una distribución exponencial de parámetro λ .

$$P(T_i < t) = 1 - e^{-\lambda t} \quad t \geq 0, \lambda \geq 0 \quad (2.1)$$

Sea un proceso estocástico X discreto, estacionario en sentido amplio (WSS), con media λ y varianza σ^2 . Los términos de esta secuencia describen el número de llegadas en un intervalo de tiempo t y se distribuyen según variable aleatoria de Poisson.

$$P(X_t = n) = \frac{(\lambda t)^n}{n!} e^{-\lambda t} \quad (2.2)$$

Las fluctuaciones horarias de la tasa de llegadas se introducen considerando el modelo de Poisson homogéneo sólo para intervalos de una hora de duración con la tasa correspondiente. Estas tasas horarias siguen patrones diarios predecibles, destacando una hora del día que presenta la mayor actividad y para la cual se realiza el dimensionamiento de los enlaces.

El segundo parámetro necesario para describir el comportamiento de los usuarios es la duración de las llamadas, que se distribuye según una variable aleatoria exponencial. La propiedad fundamental de esta distribución estadística es la falta de memoria (*memoryless*), es decir, conocer la duración de las llamadas anteriores no reduce la incertidumbre de la duración de la siguientes; analíticamente se puede expresar como:

$$P(T \leq t + t_0 | T \geq 0) = P(T \leq t)$$

2.6. Modelos de sistemas

El modelo $M/M/m/m$ corresponde en teoría de colas al siguiente esquema: central de conmutación con m circuitos de salida, usuarios descritos mediante un proceso Poisson de llegadas, y duración del servicio distribuida exponencialmente.

En este sistema, la probabilidad de rechazo de llamadas, es decir, de que

una nueva petición de servicio no encuentre ninguna línea libre, viene dada por la fórmula B de Erlang.

$$B\left(m, \frac{\lambda}{\mu}\right) = E_{1,m}\left(\frac{\lambda}{\mu}\right) = \frac{\frac{(\lambda/\mu)^m}{m!}}{\sum_{k=0}^m \frac{(\lambda/\mu)^k}{k!}} \quad (2.3)$$

Los posibles reintentos de llamada tras bloqueo se modelan tradicionalmente incrementando la tasa del proceso de llegadas, despreciando la correlación existente entre las llamadas bloqueadas y los reintentos.

Estos modelos apoyan la estructura jerárquica de la red telefónica. Se comprueba que la concentración de usuarios hacia niveles superiores de la jerarquía permite obtener la misma probabilidad de bloqueo con un número menor de enlaces; el proceso de generación de tráfico resultante se asemeja cada vez más a ruido blanco gaussiano, cuya varianza disminuye con $\frac{1}{n}$ a medida que aumenta el número de usuarios.

El dimensionamiento de los elementos de conmutación de tráfico telefónico se puede realizar en base a modelos sencillos. Dado que el ancho de banda requerido es fijo, la red se dimensiona en base a la disponibilidad, definida por la probabilidad de rechazo de llamada; mediante el dimensionamiento para el caso de mayor intensidad de tráfico (hora cargada) se logra prevenir la congestión en la red.

2.7. Modelos del tráfico de datos en LANs

Durante las décadas 80 y 90, se produjo un crecimiento exponencial del tráfico en las redes conmutadas de paquetes (RCP).

El fenómeno de interconexión de redes de área local (Internet), ha evolucionado desde sus inicios militares, pasando por el ámbito académico universitario hasta su salida al entorno comercial. Los servicios clásicos, terminal remoto, telnet, transferencia de ficheros, . . . han dado paso a una plétora de nuevas aplicaciones cuyo auge está llegando al nivel del servicio telefónico. Estos nuevos servicios y las capacidades de transmisión de las nuevas redes ópticas, han llevado al tráfico de datos a representar una parte muy importante del tráfico de información mundial.

No basta sólo con conocer y controlar el comportamiento de la red; se debe garantizar su funcionamiento y unos parámetros contratados de calidad, igual que en el servicio de telefonía.

2.8. Modelos clásicos de sistemas de colas

El modelo más simple de teoría de colas supone que el proceso de llegada de paquetes y el tiempo de servicio se distribuyen según una variable aleatoria Poisson y exponencial, respectivamente.

El tiempo de transmisión de unidades de tráfico no es exponencial, sino que presenta una clara bimodalidad debido a la presencia del protocolo TCP en el nivel de transporte. El funcionamiento de este protocolo consiste en el envío de paquetes del máximo tamaño permitido por la red (MTU) y paquetes de confirmación de datos (ACK).

El proceso de llegadas no es un proceso Poisson (se verá en los siguientes apartados) y ésto dificulta la provisión de soluciones analíticas.

2.9. *El fallo del modelo Poisson*

En los comienzos de ARPANET los resultados arrojados por los modelos poissonianos se ajustaban a las medidas de tráfico coleccionadas, pero las características del tráfico de datos actual no coinciden con las de un proceso Poisson.

Definición: 1 *Un proceso de llegada de paquetes se puede definir por el instante de llegada de estos paquetes, $a = (a_k : k = 0, 1 \dots)$, o por el tiempo transcurrido entre la llegada de dos unidades de información consecutivas,*

$T = (T_k : k = 1, 2, \dots)$.

Se define un proceso de cuenta de paquetes $X = (X_i : i = 1, 2, \dots)$ de forma que cada X_i es el número de llegadas que se producen en un intervalo de anchura δ , siendo estos intervalos contiguos.

X es un proceso de Poisson si las variables X_i son independientes e idénticamente distribuidas (i.i.d) con distribución de Poisson de parámetro λ . En este caso las variables aleatorias T_i son también variables i.i.d con distribución exponencial de parámetro λ .

2.10. Los tiempos entre llegadas no son exponenciales

Para un proceso de Poisson la marginal del proceso T es una función exponencial, y para cualquier desplazamiento no nulo la función correlación vale cero (si el desplazamiento es nulo vale 1).

[Raj Jain, 1986], [W. Leland, 1990], [Fowler and Leland, 1991] han demostrado que no se observa ninguna de estas características en el tráfico LAN (*local area network*).

[Raj Jain, 1986] distinguieron diferentes zonas en la marginal del proceso T , que les llevó a formular un modelo diferente, basado en "trenes de paquetes" (*packet train model*). En este modelo, el tráfico entre pares de nodos se da en forma de ráfagas de paquetes muy próximos entre sí. Los tiempos de separación entre ráfagas se diferencian en varios órdenes de magnitud de los tiempos de separación entre paquetes.

2.11. Ráfagas en todas las escalas

Las herramientas de monitorización han permitido conocer que el proceso de llegada de unidades de tráfico presenta ráfagas en un amplio rango de escalas temporales.

Examinando detalladamente una ráfaga de tráfico, se puede comprobar que está compuesta por periodos de baja actividad separados por unos pocos periodos muy activos. Algunas medidas de intermitencia se explican a continuación:

Relación entre el ancho de banda pico y el ancho de banda medio:

no es fácil de interpretar dada la fuerte dependencia con respecto al intervalo en el que se calculan.

Coefficiente de variación del proceso de llegadas: no converge rápidamente hacia cero a medida que aumenta el intervalo de observación, a diferencia de su comportamiento en los modelos más empleados.

Índice de dispersión de cuentas (IDC): esta relación entre la varianza del número de llegadas en un intervalo y su valor medio crece a lo largo de varios órdenes de magnitud de agregación del proceso; en los procesos Poisson se estabiliza en escalas de tiempo relacionadas con sus parámetros de definición.

Si se aumenta la capacidad del buffer de los nodos para absorber las ráfagas del tráfico real, la probabilidad de desbordamiento de los mismos **disminuye** en menor proporción que con tráfico poissoniano (sería necesario que el tamaño del buffer sobrepasara el límite impuesto por los requisitos de máximo retardo en el nodo). Se puede acotar el porcentaje de pérdidas en grandes intervalos de tiempo pero éste no se mantiene en intervalos inferiores sino que las pérdidas se concentran en los intervalos de alta actividad y desaparecen en los de baja actividad.

En definitiva, el fenómeno de ráfagas en todas las escalas de tiempo lleva a un sobredimensionamiento de la red si se busca minimizar la congestión, con la consiguiente pérdida de economía de escala.

2.12. ¿Sigue siendo válido un modelo de tráfico Poisson?

El modelo Poisson se viene empleando desde hace más de medio siglo en el estudio y diseño de redes telefónicas, debido a la tratabilidad analítica que le confieren sus propiedades matemáticas:

1. La superposición de procesos Poisson es un proceso Poisson con parámetro λ igual a la suma de sus componentes.
2. Poseen la propiedad de falta de memoria (*memoryless*), que consiste en que el tiempo que resta para la llegada del siguiente paquete es independiente del tiempo transcurrido desde la última llegada (distribución exponencial).
3. Si un tráfico Poisson se descompone aleatoriamente entre varias salidas (por ejemplo, cada paquete va destinado a una u otra salida del conmutador), entonces los procesos resultantes (el tráfico que llega a cada salida) también se distribuyen según una variable aleatoria Poisson.

Los procesos Poisson modelan de forma adecuada el proceso de llegada de conexiones TCP en servicios como telnet, ftp, no así para los servicios smtp, ftp-data o www (el tiempo transcurrido entre dos conexiones consecutivas no se distribuye según una variable aleatoria exponencial).

El tiempo entre la llegada de paquetes no se ajusta a una distribución exponencial, que subestima la capacidad que tiene el tráfico de formar

ráfagas. Los procesos derivados del de Poisson, procesos compuestos de Poisson (*compound Poisson*), procesos de llegadas en bloque de Poisson, hiperexponenciales o procesos de Poisson modulados por Markov (*Markov-modulated Poisson processes*) tampoco son capaces de capturar las características de las ráfagas que se producen en el tráfico de datos, carecen de la variabilidad suficiente para representar el tráfico real ².

2.13. Modelos tradicionales

2.13.1. Modelos On-Off Poisson

El modelo Poisson es muy rígido, únicamente permite ajustar la media. Para poder caracterizar con más exactitud los estadísticos de segundo orden del tráfico (varianza) se utiliza el modelo de tráfico: **Poisson on/off**.

Se sabe que el tráfico de datos es fuertemente impulsivo; la fuente de tráfico alterna periodos de silencio (estado *off*) con periodos de actividad (estado *on*), en los que genera datos a una tasa constante. El modelo **Poisson on/off** permite capturar esta dependencia a corto plazo. Consiste en un proceso Poisson que es activado (estado *on*) y desactivado (estado *off*) alternativamente; la duración de estos intervalos sigue una distribución exponencial de media μ_{on} y μ_{off} (respectivamente). La probabilidad de que el proceso se encuentre en estado *on* es:

$$P_{on} = \frac{\mu_{on}}{\mu_{on} + \mu_{off}} \quad (2.4)$$

X_n es un proceso que representa el estado del proceso en el instante n ; X es una variable aleatoria Bernoulli que toma el valor 1 si el proceso se encuentra en estado *on* y valor 0 si se encuentra en estado *off*.

A_n es un proceso que representa la tasa de transmisión en el instante n , $A_n = \lambda X_n$. Los valores A_n son independientes, distribuidos idénticamente, pero su distribución se permite que sea general.

$$E[A_n] = \lambda P_{on}$$

$$Var[A_n] = \lambda^2 (P_{on})(1 - P_{on})$$

Si se representa la duración media de los periodos *on/off* y la tasa media de generación de paquetes en el periodo activo por $1/\sigma_A, 1/\sigma_S$ y λ respectivamente, sus funciones de densidad de probabilidad vienen dadas por:

$$P[X \leq t] = 1 - e^{-\sigma_A t}$$

²La función de correlación de estos procesos es nula a cualquier desplazamiento, mientras que el tráfico real presenta una fuerte correlación.

$$P[Y \leq t] = 1 - e^{-\sigma s t}$$

$$P[Z \leq t] = 1 - e^{-\lambda t}$$

2.13.2. Modelos Markov

El modelo *on/off* es muy rígido, sólo reconoce dos estados de transmisión, o no se transmite nada o se genera tráfico a una tasa constante λ . Para poder caracterizar comportamientos más complejos es necesario introducir de forma explícita en el modelo la noción de estados. Ésto es lo que hacen los modelos de tráfico modulados por Markov.

Sea el espacio de estados $S = s_1, s_2, \dots, s_M$ y X_n una variable aleatoria que define el estado en el instante n . El conjunto de variables aleatorias X_n forman una cadena de Markov si la probabilidad del siguiente valor, $X_{n+1} = s_j$, depende únicamente del estado actual; ésto se conoce como **propiedad de Markov**, que implica que el futuro depende del estado actual y no de los estados previos ni del tiempo en cada estado.

En el caso de un proceso markoviano con espacio de estados discreto, el número de valores que puede adoptar X_n queda acotado a un conjunto finito, por tanto el proceso queda completamente definido por dicho conjunto y por una matriz de transición bidimensional que determina la probabilidad de transitar de un estado a otro o de quedarse en el actual.

Las principales ventajas de estos modelos son:

- Capacidad de ser aplicados a distintos tipos de tráfico (modelos de propósito general).
- Parsimonia y transportabilidad si el número de estados es reducido.
- Simplicidad de la matriz de conmutación a partir del tráfico a imitar.
- Sencillez e implementación en la medida en que las dependencias son simples (a corto plazo: de un estado con el anterior).
- Tratamiento analítico en colas bien conocido y estudiado.

Sus desventajas:

- Incrementar la complejidad del modelo (número de estados) exige un crecimiento geométrico de la parametrización del mismo para definir una matriz de conmutación bidimensional.

- Incapacidad de representar eficientemente nuevos aspectos detectados en el tráfico, especialmente la variabilidad o dependencia a largo plazo (*Long Range Dependence*). No obstante, este fenómeno tiene una repercusión relevante sólo para escenarios de transmisión con buffers grandes y ocupaciones altas, condiciones poco realistas para la transmisión de muchos tipos de servicios. Para ocupaciones bajas y buffers pequeños un modelo markoviano puede aproximar razonablemente bien la realidad. A costa de perder buena parte de su tratabilidad, si se utilizan tiempos de permanencia en los estados de corte subexponencial (distribución de Pareto), el modelo (semimarkoviano) podría sintetizar tráfico con características LRD.

2.13.3. Procesos modulados de Markov

En un proceso modulado por Markov, X_n determina en el tiempo (o modula) los parámetros de generación estadística de otro proceso estocástico Y_n , que debe poseer cierto significado físico (cantidad de información en bits, células o paquetes recibidos durante cierto intervalo o bien tiempo entre llegada de dichas unidades).

En otras palabras, en un proceso modulado por Markov, el proceso evoluciona de una manera markoviana por una cadena de estados. Para cada estado la generación de tráfico se efectúa de una manera distinta, habitualmente siguiendo una función de distribución cuyos parámetros dependen del propio estado.

- Proceso determinista modulado por Markov, MMDP (*Markov Modulated Deterministic Process*). Es un proceso determinista en el que el tiempo entre llegadas de unidades de tráfico cambia según el estado en que se encuentre el proceso de Markov asociado. El tráfico *on/off* determinista es un caso particular de MMDP.

- Proceso de Poisson modulado por Markov, MMPP (*Markov Modulated Poisson Process*). Es un proceso de Poisson en el que el parámetro λ cambia según el estado en que se encuentre el proceso de Markov asociado. El tráfico *on/off* de Poisson es un caso particular de MMPP.

Se combina la simplicidad de la conmutación markoviana con la de la generación poissoniana. Por la propia adecuación de Poisson para caracterizar el agregado de procesos estadísticamente independientes, de acuerdo con el teorema de Palm, los procesos MMPP han sido muy utilizados para representar el tráfico de fuentes multiplexadas.

El tráfico que se deriva de agregar fuentes MMPP, puede ser modelado, con total equivalencia, por otra fuente MMPP.

- Proceso de Bernoulli modulado por Markov, MMBP (*Markov Modulated Bernoulli Process*). En este caso lo que cambia según el estado en

que se encuentre el proceso de Markov es el parámetro p de un proceso de Bernoulli.

2.13.4. Modelos autoregresivos

El modelo autorregresivo AR de orden p tiene la forma:

$$X_t = \phi_1 X_{t-1} + \phi_2 X_{t-2} + \dots + \phi_p X_{t-p} + b\epsilon_t \quad (2.5)$$

donde ϵ_t es ruido blanco, los valores ϕ_j son números reales y los valores X_t corresponden a las salidas del proceso.

Si se define el operador B tal que $X_{t-1} = BX_t$, y se define el polinomio

$$\Phi(B) = 1 - \phi_1 B - \phi_2 B^2 - \dots - \phi_p B^p,$$

el proceso AR(p) se puede expresar de la siguiente forma:

$$\Phi(B)X_t = b\epsilon_t \quad (2.6)$$

La función de autocorrelación ρ_k es:

$$\rho_k = \phi_1 \rho_{k-1} + \phi_2 \rho_{k-2} + \dots + \phi_p \rho_{k-p} \quad (2.7)$$

Esta función de autocorrelación decae exponencialmente, por tanto el proceso AR(p) es incapaz de capturar las funciones de autocorrelación que decaen a una velocidad más lenta.

Para caracterizar el tráfico real es necesario encontrar el mejor ajuste para los parámetros ϕ_1, \dots, ϕ_p, b .

ϵ_t puede ser cualquier proceso estocástico, pero si es gaussiano, con media ν y varianza σ^2 , es suficiente conocer el valor de tres parámetros para estimar el rendimiento de la cola.

Así se reduce la complejidad de encontrar todos los parámetros requeridos para un modelo autorregresivo de orden k y utilizar únicamente el proceso autorregresivo de primer orden.

$$X_t = aX_{t-1} + b\epsilon_t \quad (2.8)$$

2.13.5. Modelos autoregresivos con media móvil

El modelo ARMA de orden (p, q) tiene la forma:

$$X_t = \phi_1 X_{t-1} + \phi_2 X_{t-2} + \dots + \phi_p X_{t-p} + b\epsilon_t - \theta_1 \epsilon_{t-1} - \theta_2 \epsilon_{t-2} - \dots - \theta_q \epsilon_{t-q} \quad (2.9)$$

Si se define B y $\Phi(B)$ como en el apartado anterior y $\Theta(B) = 1 - \theta_1 B - \theta_2 B^2 - \dots - \theta_q B^q$ el modelo se puede representar como

$$\Phi(B)X_t = \Theta(B)\epsilon_t \quad (2.10)$$

La función de autocorrelación del proceso ARMA(p, q) se calcula para todo k , pero para $k > q$ la autocorrelación del modelo ARMA(p, q) coincide con el del proceso AR(p). ARMA(p, q) es incapaz de representar procesos con funciones de autocorrelación que decaen subexponencialmente.

2.13.6. Modelos autoregresivos integrados de media móvil

El modelo ARIMA de orden (p,d,q) es una extensión del modelo ARMA (p,q) y el polinomio $\Phi(B)$ tiene d raíces igual a la unidad. El modelo ARIMA de orden (p,d,q) tiene la forma:

$$\Psi(B)\Delta^d X_t = \Theta(B)\epsilon_t \quad (2.11)$$

donde Δ es el operador diferencia definido como

$$\Delta X_t = X_t - z_{t-1} = (1 - B)X_t$$

Cuando d es un número entero, el proceso ARIMA (p,d,q) es estrictamente SRD (*short range dependent*). Sin embargo, si d es un número real se puede extender a LRD.

La razón principal de la resistencia de los modelos tradicionales es la ausencia de respuesta a las siguientes cuestiones:

- ¿Cuál es la explicación física de la naturaleza auto-similar del tráfico en las redes de comunicaciones?
- ¿Cuál es el impacto de la auto-similitud en las redes, diseño de protocolos y análisis de prestaciones?

Se puede dar respuesta a la primera pregunta argumentando los resultados matemáticos apropiados y analizando detalladamente los estadísticos de las trazas del tráfico Ethernet a escalas pequeñas.

2.14. Modelado fractal de tráfico

En los últimos años se han realizado varios estudios estadísticos sobre trazas de tráfico tomadas en entornos LAN (redes Ethernet) [Leland et al., 1993], redes de área extensa [Paxson, 1995] o redes de distribución de vídeo digital [Heyman and Lakshman, 1996].

Estos análisis han puesto de manifiesto la **naturaleza fractal o autosemejante del tráfico**.

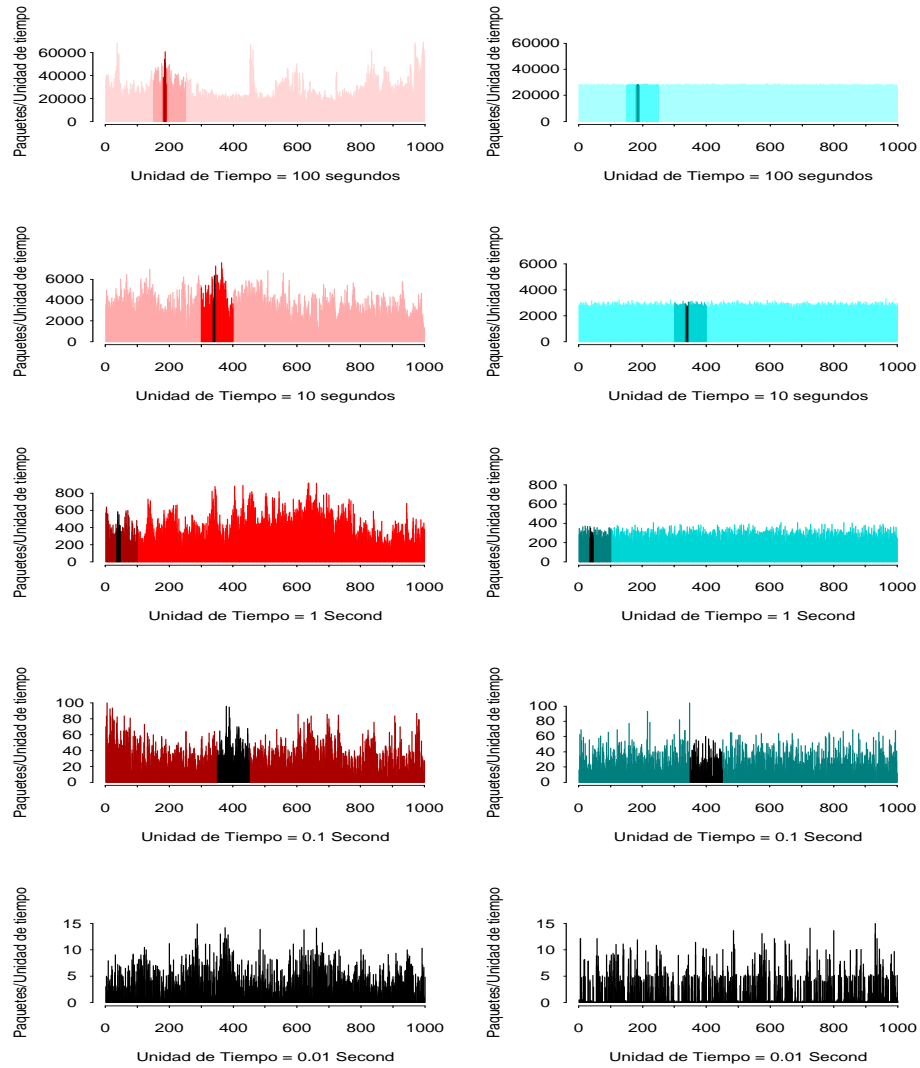
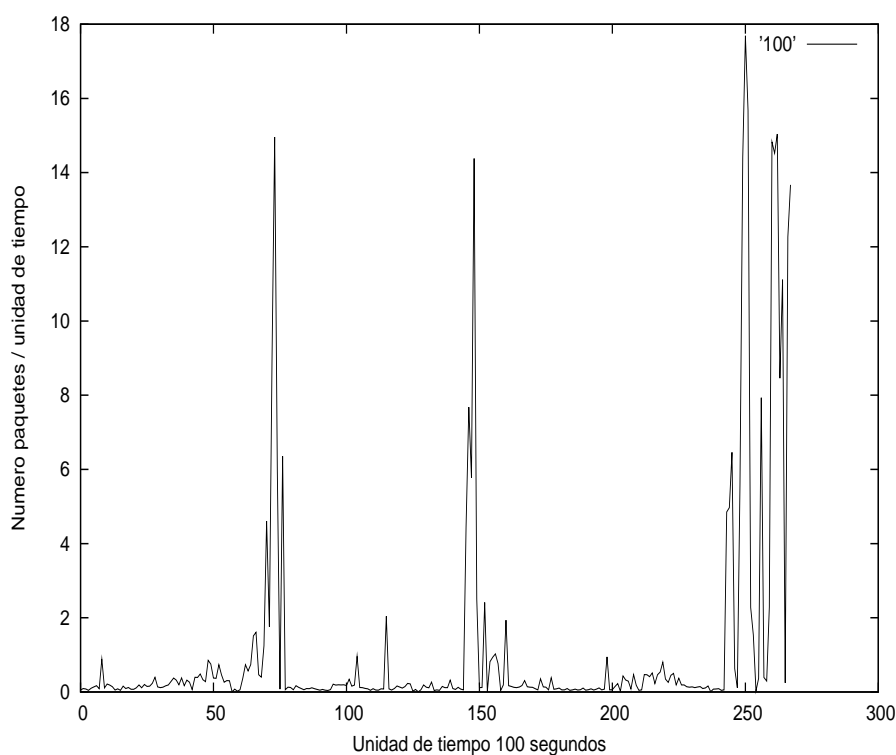


Figura 2.1: Tráfico real (izquierda)

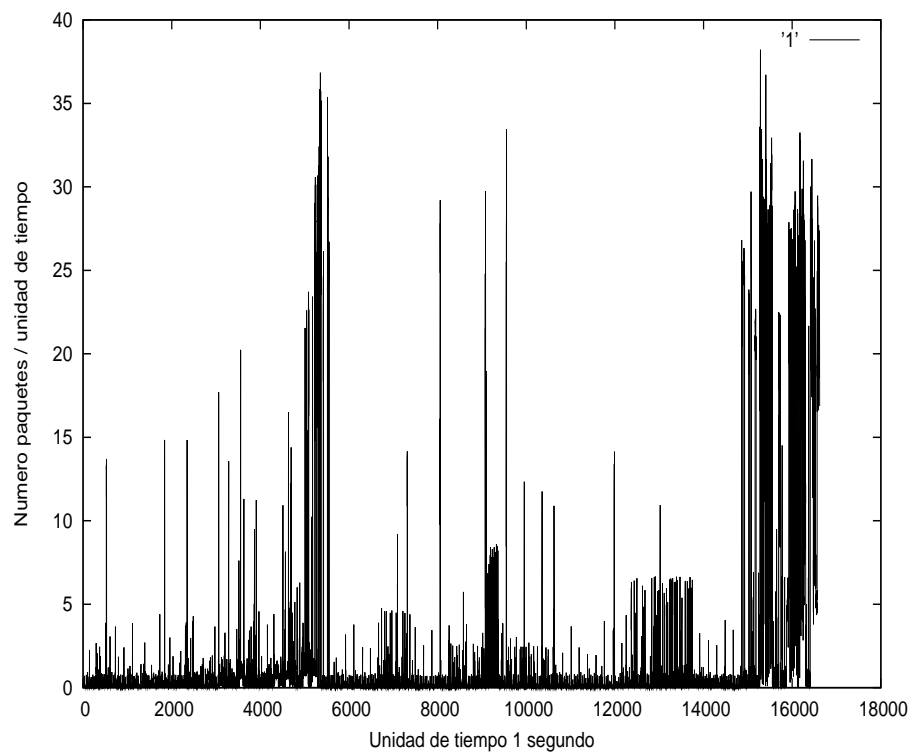
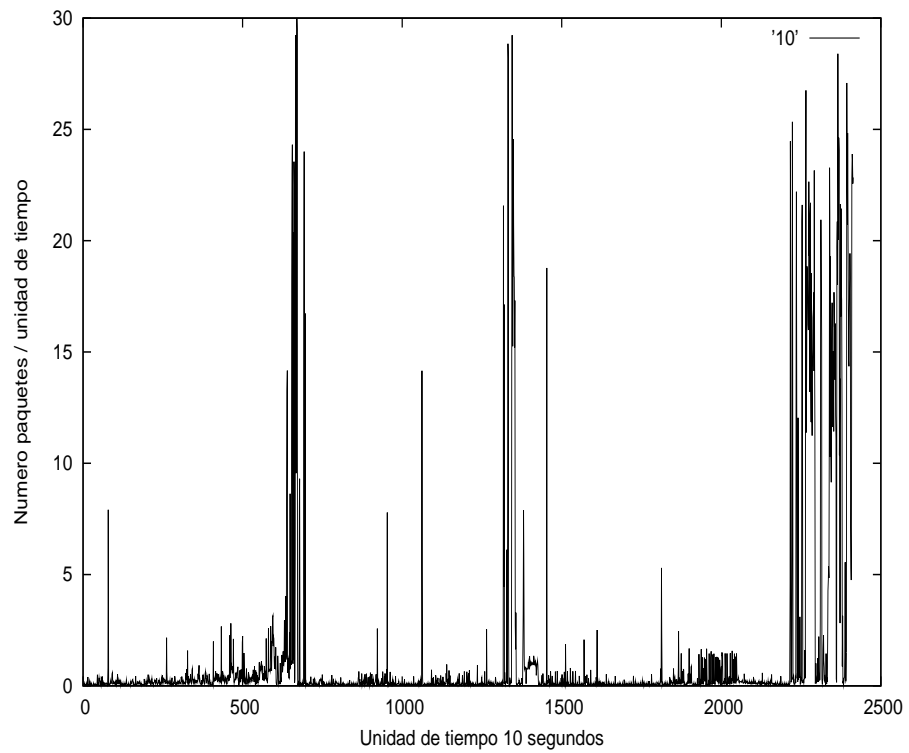
Imitando el trabajo de [Leland et al., 1993] (ver fig 2.1), y para ilustrar este fenómeno en la red Ethernet del departamento de Ingeniería Telemática, se representa gráficamente la tasa de tráfico monitorizado en distintas escalas de tiempo utilizando la herramienta **gnuplot** (ver apéndice de herramientas).

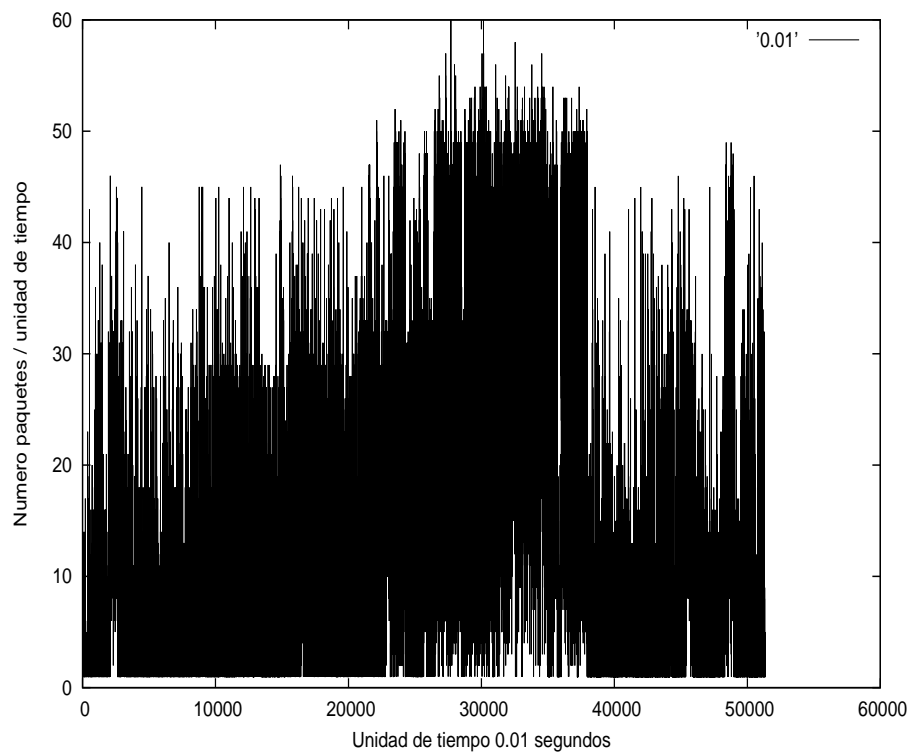
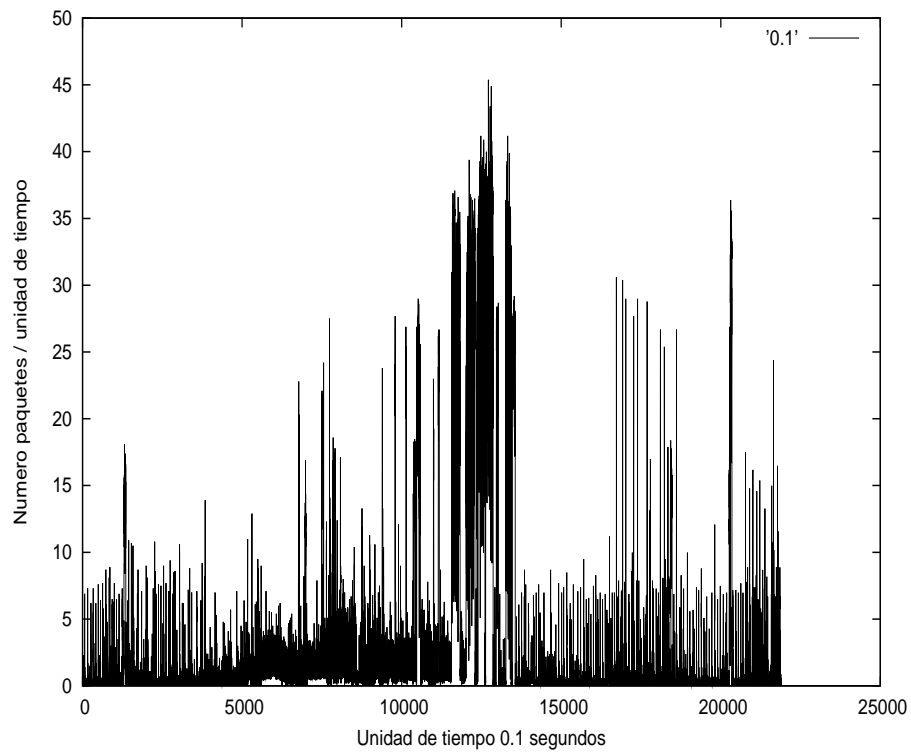
Para un periodo de 15 horas consecutivas de tráfico monitorizado en el segmento de red de Ingeniería Telemática (tráfico interno y externo), las siguientes figuras representan la tasa de llegada de paquetes (número de paquetes por unidad de tiempo) para cinco escalas de tiempo diferentes. Empezando con la escala de tiempo mayor (100 segundos (a)), la siguiente gráfica se obtiene incrementando la resolución temporal por un factor de 10 y concentrándose en un subintervalo elegido aleatoriamente ³.

Todas las representaciones gráficas parecen, de forma intuitiva y en un sentido de distribución estadística, muy similares; un segmento del proceso en una escala de tiempo se comporta de la misma forma que la versión escalada del proceso en otra escala de tiempo diferente (comportamiento auto-semejante o fractal). El carácter intermitente de la señal representada se mantiene en todas las escalas de tiempo de interés (las ráfagas no tienen un tamaño más o menos fijo, sino que su duración oscila entre unos pocos milisegundos y algunas horas).



³La unidad de tiempo correspondiente a la escala temporal más pequeña es 10 milisegundos





La representación gráfica del tráfico generado por modelos clásicos (ver Figura 2.1) no mantiene una estructura semejante en las distintas escalas de tiempo, sino que se aplanan dos o tres órdenes de magnitud más allá. Dicho de otro modo, los modelos clásicos son capaces de capturar únicamente la dependencia del tráfico a corto plazo (ráfagas en el orden de milisegundos). Se puede decir que los modelos clásicos no son válidos para caracterizar el tráfico de la red bajo estudio.

Los análisis presentados en este trabajo ponen de manifiesto que las trazas de tráfico corresponden a las observaciones de un proceso asintóticamente auto-similar de segundo orden o dependiente a largo plazo.

En definitiva, el modelado de tráfico de la red de Ingeniería Telemática requiere la utilización de procesos estocásticos fractales. Existen numerosos modelos matemáticos de tráfico auto-similar o dependiente a largo plazo. Algunos facilitan el análisis de colas, otros son motivados físicamente y otros muestran que la dependencia a largo plazo puede ser generada de distintas formas. La abundancia de modelos matemáticos puede distraer de una importante característica, los mecanismos físicos y causales subyacentes en el fenómeno de red incluyendo las características del tráfico. Los modelos de tráfico físicos que remontan las raíces de la auto-similitud y la dependencia a largo plazo a propiedades tales como los protocolos de red y distribución del tamaño de los ficheros en los servidores, tienen una clara ventaja con respecto a la predicción y verificación sobre los modelos de "caja negra" asociados con los análisis de series temporales tradicionales.

En cualquier caso el éxito de los modelos fractales viene determinado por las consecuencias prácticas que posee la existencia de la LRD (*long range dependence*) en el tráfico telemático. Así, una de los principales efectos de cargar un multiplexor con tráfico fractal, es un decaimiento lento de las pérdidas con el aumento del tamaño del buffer de contención. Esto implica la inutilidad de las técnicas de **buffering** cuando el tráfico es auto-similar, ya que un aumento del tamaño del buffer supone un crecimiento del retardo sin obtener ganancias significativas en las pérdidas.

Otra problemática asociada al carácter fractal del tráfico es su escasa tratabilidad analítica así como la convergencia más lenta de las medidas de simulación que imponen estos procesos de variabilidad a largo plazo (las simulaciones con procesos fractales pueden llegar a ver su duración incrementada en varios órdenes para obtener resultados con la misma fiabilidad que los que se pueden conseguir con procesos SRD o incorrelados similares).

La investigación de la auto-similitud del tráfico se clasifica en cuatro categorías:

Modelado basado en las medidas de tráfico se centra en la colección y análisis de las medidas (de alta calidad y diversidad) de tráfico LAN para detectar, identificar y cuantificar las características pertinentes. No solo se ha trabajado en la caracterización del tráfico sino también en el área de la inferencia estadística y científica, que ha sido esencial para la detección y cuantificación de la auto-similitud o dependencia a largo plazo.

En un nivel formal, la validez de una técnica de inferencia o estimación está sujeta a un proceso subyacente que presumiblemente generó los datos. Así que, en general, un conjunto de datos de origen desconocido no puede ser únicamente atribuido a un modelo específico; el principal (y único) propósito de la inferencia estadística o científica es tratar con este problema concluyendo si los datos son consistentes con una estructura de modelo asumida. Ser consistente con un modelo asumido no descarta la existencia de otros modelos que conformarían los datos igualmente bien.

En un nivel práctico, muchas de las técnicas de inferencia comúnmente usadas para cuantificar el grado de auto-similitud o dependencia a largo plazo exhiben diferentes idiosincrasias y propiedades de robustez. Debido a su naturaleza heurística, estas técnicas son generalmente fáciles de usar y aplicar, pero los resultados son difíciles de interpretar.

Modelado físico explica las causas físicas de la auto-similitud en el tráfico de red (fenómeno empíricamente observado) considerando las propiedades de los sistemas distribuidos. Asimismo provee nuevas ideas acerca de la naturaleza dinámica del tráfico. Propiedad empírica de los sistemas distribuidos: la distribución heavy-tailed del tamaño de los ficheros u objetos. Si los hosts intercambian ficheros cuyos tamaños son heavy-tailed, el tráfico de red resultante en los puntos de multiplexación en la capa de red es auto-similar. Este fenómeno causal es robusto en el sentido de mantenerse para una variedad de protocolos de la capa de transporte y para distintas configuraciones de red. Esta causalidad estructural no tendría sentido a menos que se pudiera explicar por qué los objetos heavy-tailed transportados en la red inducen auto-similitud en los puntos de multiplexación. El modelo *on/off* establece que la superposición de un gran número de fuentes independientes on/off con periodos on/off heavy-tailed conduce a la auto-similitud en el proceso agregado - un proceso FGN - cuya dependencia a largo plazo es determinada por la pesadez de la cola (heavy tailedness) de los periodos on/off. La agregación en el espacio no es esencial para inducir dependencia a largo plazo - ésto es responsable

de la propiedad gaussiana del tráfico agregado por una aplicación del teorema central del límite. Sin embargo, es relevante para describir el tráfico de red multiplexado. El modelo on/off tiene sus raíces en un renewal reward process presentado por Mandelbrot y provee la underpinning teórica para gran parte del trabajo en el modelado físico del tráfico de red. Ésto junto con la evidencia empírica de las duraciones on/off heavy-tailed, representa una explicación directa y de más bajo nivel de las razones físicas de la auto-similitud y forma los principales factores que distinguen el modelo on/off de otros modelos matemáticos de tráfico auto-similar.

Análisis de colas provee modelos matemáticos de tráfico dependiente a largo plazo para facilitar los análisis de rendimiento.

El comportamiento de un buffer con entrada dependiente a largo plazo exhibe características de rendimiento diferentes de los sistemas correspondientes con entradas markovianas.

La distribución de la longitud de la cola decae subexponencialmente si el tráfico de entrada es auto-similar mientras que para entradas markovianas el decaimiento es exponencial. Ésto implica que el buffering - como estrategia de provisión de recursos - resulte inefectivo cuando el tráfico de entrada es auto-similar por incurrir en un retardo de cola desproporcionado respecto a reducción en la tasa de pérdida de paquetes.

Ésto ha llevado a una estrategia de provisión de recursos *baja capacidad de buffer / gran ancho de banda* por su influencia simple y limitante en la cola: si la capacidad del buffer se reduce, disminuye la capacidad de encolar. Además, cuanto más pequeña es la capacidad del buffer, más relevantes se convierten las correlaciones a corto plazo para determinar la ocupación del buffer.

Un impacto del análisis de rendimiento basado en colas ha sido la adopción creciente del paradigma del dimensionamiento de recursos, que establece que la capacidad del buffer debería decrecer mientras que el ancho de banda del enlace debería aumentar. Cuando la estructura de correlación es débil, un incremento de la capacidad del buffer no supone un incremento en su ocupación. Ésto sí ocurre cuando la estructura de correlación es fuerte.

Con respecto a las medidas de rendimiento de primer orden, tales como tasa de pérdida de paquetes se convertirían en factores dominantes.

Control de tráfico y provisión de recursos se puede separar en provisión de recursos y dimensionamiento.

El análisis de colas con entrada auto-similar tiene que ver directamente con el problema de dimensionamiento de recursos. La cuestión de estimar cuantitativamente la utilidad marginal de una unidad de

recurso adicional tal como el ancho de banda o capacidad del buffer es respondida, en parte, con la ayuda de esta técnica. Del lado de control de reacción está el trabajo sobre control de congestión en múltiples escalas de tiempo, que intenta aprovechar la estructura de correlación que existe a lo largo de las múltiples escalas de tiempo en el tráfico auto-similar para propósitos de control de congestión. A pesar del impacto negativo de la auto-similitud en el rendimiento, la dependencia a largo plazo admite la posibilidad de utilizar la correlación a grandes escalas de tiempo, transformando esto último para aprovechar la estructura de precitabilidad, que puede ser afectada para guiar las acciones de control de congestión a pequeñas escalas de tiempo para producir ganancias significativas en el rendimiento.

Procesos auto-similares en redes de comunicaciones

Leland sugirió el uso de modelos parsimoniosos para el tráfico auto-similar; estos modelos deben ser matemáticamente tratables y físicamente razonables .

A continuación se van a presentar las definiciones y propiedades de los procesos auto-similares discretos en el tiempo, complementadas con algunas condiciones más generales de auto-similitud. Los procesos autosimilares resultan muy interesantes dado que capturan de forma simple características importantes del tráfico Ethernet como la presencia de dependencia a largo plazo.

3.1. Algunos procesos auto-similares

En este apartado se van a definir dos procesos de este tipo:

- FGN (*Fractional Gaussian Noise*)
- FARIMA (*Fractional Autoregressive Integrated Moving-Average*).

Estos procesos facilitan el modelado parsimonioso de series temporales dependientes a largo plazo.

Los modelos tradicionales (ARMA - *AutoRegressive Moving-Average* -, ARIMA - *Autoregressive Integrated Moving-Average* -) sólo son capaces de modelar una porción a corto plazo de las correlaciones del conjunto de datos.

FGN con parámetro $H \in (0,1)$ es un proceso gaussiano estacionario con media μ , varianza σ^2 , y función de autocorrelación:

$$r(k) = \frac{1}{2}((k+1)^{2H} - 2k^{2H} + |k-1|^{2H}), \quad k > 0$$

Su espectro de potencia es definido por:

$$f(\lambda) = \frac{\sigma^2}{\pi} \text{sen}(\pi H) \Gamma(2H+1) (1 - \cos \lambda) \sum_{j=-\infty}^{\infty} |\lambda + 2\pi j|^{-2H-1}$$

También puede ser definido como el proceso que tiene la misma función de correlación que el proceso de incrementos unidad $\Delta B_H(t) = B_H(t) - B_H(t-1)$ del proceso FBM con exponente H.

La caracterización completa de un proceso FARIMA requiere tres parámetros: p,d,q. Los órdenes p y q son los parámetros clásicos de un proceso ARMA y $d = H - 0,5$ es el parámetro de diferencia fractal.

La definición de un proceso FARIMA(0,1,0) es

$$(1 - B)X_t = a_t \quad (3.1)$$

donde B denota el operador desplazamiento y a_t es un proceso de ruido blanco, pero se puede generalizar para un proceso FARIMA(0,d,0):

$$(1 - B)^d X_t = a_t \quad -0,5 < d < 0,5 \quad (3.2)$$

El operador diferencia fractal, $(1 - B)^d$, es definido por la serie binomial

$$(1 - B)^d = \sum_{k=0}^{\infty} \binom{d}{k} (-B)^k$$

El espectro de potencia es:

$$f(\lambda) = (2\text{sen}\lambda/2)^{-2d} = |1 - e^{i\lambda}|^{-2d}, \quad 0 < \lambda \leq \pi \quad (3.3)$$

Por tanto, $f(\lambda) \sim \lambda^{-2d} = \lambda^{1-2H}$, $\lambda \rightarrow 0$.

Los procesos FARIMA(p,d,q) con valores de p y/o q distintos de 0 son capaces de aproximar adecuadamente la dependencia a largo plazo (*long range dependence*) y la dependencia a corto plazo (*short range dependence*).

3.2. Procesos auto-similares continuos en el tiempo

Una definición habitual de los procesos estocásticos auto-similares se basa en un escalado de la variable aleatoria continua en el tiempo.

Un proceso estocástico $X(t)$ es estadísticamente auto-similar con parámetro H ($0,5 \leq H \leq 1$) si para cualquier número real $a \geq 0$, el proceso $a^{-H}x(at)$ tiene las mismas propiedades estadísticas que $x(t)$. Esta relación se puede expresar con estas condiciones:

1. $E[x(t)] = \frac{E[x(at)]}{a^H}$ Media
2. $Var[x(t)] = \frac{Var[x(at)]}{a^{2H}}$ Varianza
3. $R_x[t, s] = \frac{R_x(at, as)}{a^{2H}}$ Autocorrelación

El parámetro H (parámetro de Hurst o parámetro de auto-similitud), es una medida clave de la auto-similitud del proceso, de la longitud de la dependencia a largo plazo de un proceso estocástico. A medida que el valor de H se aproxima a 1 aumenta el grado de persistencia de la dependencia a largo plazo (H = 0.5 indica ausencia de auto-similitud).

Como ejemplo de esta definición, se considera el proceso FBM (*Fractional Brownian Motion*), que es utilizado frecuentemente en el análisis de tráfico de datos de auto-similar.

Un proceso FBM $B_H(t)$ se define como:

$$B_H(t) = Xt^H (t \geq 0; \quad 0,5 \leq H \leq 1)$$

donde X es una variable aleatoria normalmente distribuida con media cero y varianza 1, y H es un parámetro del proceso. Así pues, el valor del proceso en el instante de tiempo t es igual al valor de la variable aleatoria X multiplicado por el intervalo de tiempo elevado a H.

- $E[B_H(t)] = 0$
- $Var(aX) = a^2 Var(X)$
- $Var[B_H(t)] = Var[t^H X] = t^{2H}$

Estas observaciones demuestran que para H=0.5 el proceso FBM se reduce a un proceso BM normal.

La densidad de probabilidad de un proceso BM *Brownian motion* tiene la forma:

$$f_{B_H(x,t)} = \frac{1}{\sqrt{2\pi t^{2H}}} e^{-x^2/2t^{2H}}$$

La varianza de los incrementos estacionarios de este proceso es:

$$Var[B_H(t) - B_H(s)] = E[(B_H(t) - B_H(s))^2] = |t - s|^2$$

La función de correlación de $B_H(t)$ es:

$$R_{B_H(t,s)} = E[(B_H(t)B_H(s))^2]$$

Este valor se deriva de la siguiente forma:

$$E[(B_H(t) - B_H(s))^2] = E[B_H(t)^2 + B_H(s)^2 - 2B_H(t)B_H(s)]$$

Reajustando,

$$\begin{aligned} E[B_H(t)B_H(s)] &= \frac{1}{2}(E[B_H(t)^2] - E[B_H(s)^2] - E[(B_H(t) - B_H(s))^2]) \\ &= \frac{1}{2}(Var[B_H(t)] + Var[B_H(s)] - Var[B_H(t) - B_H(s)]) \\ &= \frac{1}{2}(t^{2H} + s^{2H} - |t - s|^{2H}) \end{aligned}$$

Si $H = 0,5$ la expresión anterior se reduce a la función de correlación de un proceso BM (*Brownian motion*).

Si el proceso FBM es auto-similar, debe satisfacer las tres condiciones mencionadas anteriormente.

Se considera el proceso $B_H(at)$, donde $B_H(t)$ es un proceso FBM.

- $B_H(at) = X(at)^H$ y se puede comprobar fácilmente que $E[B_H(at)] = a^H E[B_H(t)] = 0$, satisfaciendo la primera condición.
- $Var[B_H(at)] = Var[X(at)^H] = (at)^{2H} Var[X] = (at)^{2H}$. Por tanto, $Var[B_H(t)] = Var[B_H(at)]a^{2H}$, satisfaciendo la segunda condición.
- La tercera condición también es satisfecha:

$$\begin{aligned} R_{B_H}(at, as) &= \frac{1}{2}((at)^{2H} + (as)^{2H} - |at - as|^{2H}) \\ &= \frac{a^{2H}}{2}(t^{2H} + s^{2H} - |t - s|^{2H}) \\ &= a^{2H} R_B(t, s) \end{aligned}$$

Una diferencia llamativa entre el proceso BM (*brownian motion*) y el proceso FBM (*Fractional Brownian Motion*) es la relación entre sus intervalos; En el primero son independientes e incorrelados (ésto se puede demostrar considerando la correlación entre incrementos) y en el segundo no. Se considera la correlación entre los incrementos $[-t, 0]$ y $[0, t]$.

$$\begin{aligned} E[(B_H(0) - B_H(-t))(B_H(t) - B_H(0))] \\ &= \frac{-1}{2}((-t)^{2H} - t^{2H} - |-t - t|^{2H}) \\ &= \frac{1}{2}((2t)^{2H} - t^{2H}) \end{aligned}$$

Si $H = 0.5$, la correlación entre incrementos pasados y futuros es nula (incrementos independientes). Sin embargo, para $H > 0.5$ se observa el fenómeno de **persistencia**; un incremento positivo (o negativo) en algún instante de tiempo implica un nuevo incremento (o decremento) en el futuro.

Este comportamiento persistente está en conflicto con lo que normalmente se ha asumido sobre el fenómeno estocástico. En algunos procesos (BM - *brownian motion*), los incrementos son independientes; en otros la correlación se mantiene para intervalos de tiempo pequeños, Δt .

3.3. Procesos auto-similares discretos en el tiempo

En este apartado se van a presentar las definiciones de procesos exacta y asintóticamente auto-similares.¹

La naturaleza del tráfico de la red de Ingeniería Telemática es discreta (transmisión de tramas). Por tanto, el interés de este trabajo se va a centrar en modelos de tráfico discretos en valores y en tiempo.

Se va a introducir una condición necesaria y suficiente de auto-similitud en términos de la varianza de las versiones agregadas del proceso original. También se va a presentar una condición suficiente en términos de la función de correlación del proceso.

Sea un proceso estocástico discreto en el tiempo o serie temporal $X(t), t \in \mathbf{Z}$.

- $\mu = E[X(t)]$.
- $\sigma^2 = E[(X(t) - \mu)^2] \forall t \in \mathbf{Z}$, y $\mu = 0$.
- $r(-k) = r(k) = \frac{(E(X_{t+k}) - \mu)(E(X_t) - \mu)}{\sigma^2} \quad k \in I_0 = 0, 1, 2, \dots$

La media μ , la varianza σ^2 , y la función de correlación $r(k)$ son independientes del tiempo.

¹Este conjunto es demasiado restrictivo para modelar un tráfico de red real, normalmente se emplea para propósitos teóricos.

3.3.1. Auto-similitud exacta de segundo orden

Las dos definiciones que se van a introducir son equivalentes.

Definición A

Un proceso X es exactamente auto-similar de segundo orden (es-s) con parámetro $H = 1 - \frac{\beta}{2}$ si su función de correlación es:

$$r(k) = \frac{1}{2}[(k+1)^{2-\beta} - 2k^{2-\beta} + (k-1)^{2-\beta}] = g(k) \quad k \in \mathbf{N}$$

La función $g(k)$ puede ser escrita como $g(k) = \delta^2(k^{2-\beta})$ en términos del operador $\delta^2(f(x))$ aplicado a la función $f(x) = x^{2-\beta}$.

Definición B

Para la presentación de la segunda definición, es necesario introducir el proceso agregado (en bloques de longitud m), $X^{(m)}$.

$$X^{(m)} = (X_1^{(m)}, X_2^{(m)}, \dots), X_i^{(m)} = \frac{1}{m} (X_{im-m+1} + \dots + X_{im}^{(m)}), \quad m, t \in \mathbf{N}$$

Su función de correlación $r^m(k)$ y su varianza $Var(X^m)$.

Un proceso X es exactamente auto-similar de segundo orden (es-s) con parámetro $H = 1 - \frac{\beta}{2}$ si la función de correlación del proceso agregado ($r^m(k)$), coincide con el de la serie original $\forall k, m \in \mathbf{N}$.

Teorema 1 *Para un proceso X exactamente auto-similar y $0 < \beta < 1$, las siguientes propiedades son equivalentes:*

- Según la definición B:

$$r_m(k) = r(k)$$

Un proceso X con $r(k) = g(k)$ no cambia sus características de segundo orden (correlación y forma del espectro), aunque la muestra sea dividida en bloques de longitud m .

- Según la definición A:

$$r(k) = g(k) = \frac{1}{2}(2 - \beta)(1 - \beta)k^{-\beta} = H(2H - 1)k^{-\beta}, \quad k \rightarrow \infty$$

La función de correlación de la serie original decae de forma hiperbólica, no exponencial. Aunque los coeficientes de correlación para altos valores del retardo k son pequeños, el efecto acumulado es muy importante y da características que son drásticamente diferentes respecto a los procesos dependientes a corto plazo.

$$\sum_{k=0}^{\infty} r(k) = \infty$$

- $V_m \sigma^2 = m^{-\beta}$.
- $f(\lambda) = c|e^{2\pi/\lambda} - 1|^2 \sum_{i=-\infty}^{\infty} \frac{1}{|\lambda+1|^{3-\beta}}, \quad -\frac{1}{2} \geq \lambda \leq \frac{1}{2}$.

Además de los **procesos exactamente auto-similares de segundo orden** existen los **procesos estrictamente auto-similares (ss-s)**, que son más ampliamente conocidos. A continuación se va a realizar una comparación de ambos procesos.

Definición 1 *Un proceso estacionario en sentido amplio $X = (X_1, X_2, \dots)$ es **estrictamente auto-similar (ss-s)** con parámetro $H = 1 - \frac{\beta}{2}$, si*

$$m^{1-H} X^{(m)} = X$$

o equivalentemente,

$$\tilde{X}^{(m)} = X$$

$\tilde{X}^{(m)} = (\tilde{X}_1^{(m)}, \tilde{X}_2^{(m)}, \dots)$ es el proceso $X^{(m)}$ normalizado y $\tilde{X}_i^{(m)} = \frac{1}{m^H}(X_{im-m+1} + \dots + X_{im})$.

La comparación de los procesos **exactamente auto-similares de segundo orden (es-s)** y los procesos **estrictamente auto-similares de segundo orden (ss-s)** se expresa en términos de las siguientes relaciones:

$$[r_m(k) = \tilde{r}_m(k) = r(k)] \Leftrightarrow [\tilde{X}^{(m)} = X] \Rightarrow [\text{var} \tilde{X}_i^{(m)} = \sigma^2 = \text{var} X_i] \quad (3.4)$$

$$[r_m(k) = \tilde{r}_m(k) = r(k)] \Leftrightarrow [\text{var} \tilde{X}_i^{(m)} = \sigma^2] \Leftrightarrow [r(k) = g(k)] \Leftrightarrow [\text{var} X_i^{(m)} = \sigma^2 m^{-\beta}] \quad (3.5)$$

donde $\tilde{r}^{(m)}$ es la función de correlación del proceso normalizado $\tilde{X}^{(m)}$.

Estas relaciones demuestran que cualquier proceso ss-s tiene $r(k) = g(k)$. Esto significa que **si X ss-s \Rightarrow es-s**. La implicación contraria se cumple sólo si la distribución del proceso X es gaussiana.²

[X es un proceso gaussiano ss-s de media nula]

$$[r_m(k) = \tilde{r}_m(k) = r(k) = g(k)], [\text{var} \tilde{X}_i^{(m)} = \sigma^2], [\text{var} X_i^{(m)} = \sigma^2 m^{-\beta}] \quad (3.6)$$

[X es un proceso gaussiano de media nula y $r(k)=g(k)$]

$$[r_m(k) = \tilde{r}_m(k) = r(k) = g(k)], [\text{var} \tilde{X}_i^{(m)} = \sigma^2], [\text{var} X_i^{(m)} = \sigma^2 m^{-\beta}] \quad (3.7)$$

[X es un proceso gaussiano de media nula y $\text{var} \tilde{X}_i^{(m)} = \sigma^2$ (o equivalentemente, $\text{var} X_i^{(m)} = \sigma^2 m^{-\beta}$)]

$$[r_m(k) = \tilde{r}_m(k) = r(k) = g(k)], [\tilde{X}^{(m)} = X] \quad (3.8)$$

[X es un proceso gaussiano de media nula y $\tilde{r}_m(k) = r(k)$]

$$[\tilde{X}^{(m)} = X], [r_m(k) = \tilde{r}_m(k) = r(k) = g(k)], [\text{var} \tilde{X}_i^{(m)} = \sigma^2], [\text{var} X_i^{(m)} = \sigma^2 m^{-\beta}] \quad (3.9)$$

Las implicaciones anteriores muestran la equivalencia de las siguientes afirmaciones:

1. Proceso gaussiano de media nula es ss-s
2. $r_m(k) = \tilde{r}_m(k) = r(k)$
3. $\text{var} \tilde{X}_i^{(m)} = \sigma^2$
4. $r(k) = g(k)$

3.3.2. Auto-similitud asintótica de segundo orden

$\{X_n\}$ es asintóticamente auto-similar (de segundo orden) con parámetro $H = 1 - \frac{1}{2}$, si y sólo si:

$$\lim_{m \rightarrow \infty} r^{(m)}(k) = g(k), \quad k = 1, 2, 3, \dots$$

Es decir, $\{X_n\}$ es asintóticamente auto-similar si la función de correlación de $\{X_n^{(m)}\}$ se comporta (para valores grandes de m) como la de un proceso

²Un proceso ss-s no puede tener media nula mientras que la media de un proceso es-s sí puede ser distinta de cero

exáctamente auto-similar de segundo orden.

Se puede demostrar [Tsybakov, 1997] que en un proceso asintóticamente auto-similar de segundo orden, $r(k) \sim ck^{-\beta}$ cuando $k \rightarrow \infty$, siendo c una constante necesariamente igual a $H(2H - 1)$. Por este motivo, algunos autores [Pappu, 1996] definen proceso auto-similar como un proceso LRD cuya función de correlación decae hiperbólicamente.

3.4. Propiedades del tráfico autosemejante

Las tres propiedades más importantes del tráfico auto-similar son:

- Dependencia a largo plazo
- Efecto Hurst
- Decaimiento lento de la varianza

Estas propiedades, conocidas como propiedades fractales [Willinger et al., 1996] han sido observadas en el tráfico de la red de Ingeniería Telemática y no pueden ser explicadas utilizando los modelos clásicos de tráfico.

3.5. Dependencia a largo plazo y efecto Hurst

Un proceso estocástico que satisface la relación

$$r(k) \sim k^{-\beta} L_1(k), \quad k \rightarrow \infty \quad (3.10)$$

se dice que exhibe *long range dependence*.

Los procesos con dependencia a largo plazo se caracterizan por un decaimiento hiperbólico de los coeficientes de correlación según aumenta el retardo k ; en los procesos dependientes a corto plazo el decaimiento es exponencial,

$$r(k) \sim \rho^k, \quad k \rightarrow \infty \quad (3.11)$$

resultando en una función de autocorrelación sumable $0 < \sum_k r(k) < \infty$.

Cuando se trabaja en el dominio de la frecuencia, la dependencia a largo plazo se manifiesta en una densidad espectral que obedece un comportamiento potencial cerca del origen. De hecho, equivalentemente a la ecuación 3.10 (bajo débiles condiciones de regularidad en la función de lento decaimiento L_1), hay dependencia a largo plazo en X si:

$$f(\lambda) \sim \lambda^{-\gamma} L_2(\lambda) \quad \lambda \rightarrow 0, \quad 0 < \gamma < 1 \quad (3.12)$$

donde L_2 es una función que varía lentamente cerca del origen y $f(\lambda) = \sum_k r(k) e^{jk\lambda}$ denota la densidad espectral de potencia.

La dependencia a largo plazo implica que $f(0) = \sum_k r(k) = \infty$ (la densidad espectral de un proceso LRD tiende a ∞ a medida que λ se aproxima a 0 (**1/f noise**)). Por su parte, la dependencia a corto plazo se caracteriza por una función de densidad espectral $f(\lambda)$ que, en $\lambda = 0$, es positiva y finita.

Históricamente, la importancia de los procesos auto-similares subyace en el hecho de que proveen una explicación elegante y una interpretación de una ley empírica conocida como *ley de Hurst* o *efecto Hurst*.

Para un conjunto de observaciones $(X_k : k = 1, 2, \dots, n)$ con media muestral $\bar{X}(n)$ y varianza muestral $\bar{S}(n)^2$, el estadístico R/S (*rescaled adjusted range*) es dado por:

$$R(n)/S(n) = 1/S(n)[\max(0, W_1, W_2, \dots, W_n) - \min(0, W_1, W_2, \dots, W_n)] \quad (3.13)$$

con $W_k = (X_1 + X_2 + \dots + X_k) - k\bar{X}(n)$, $k = 1, 2, \dots, n$. Hurst encontró que muchas series temporales de la naturaleza satisfacían la siguiente relación:

$$E[R(n)/S(n)] \sim cn^H \quad n \rightarrow \infty \quad (3.14)$$

donde H es el parámetro de Hurst H y c es una constante positiva y finita. Si las observaciones X_k vienen de un modelo dependiente a corto plazo, Mandelbrot y Van Ness (1968) demostraron que:

$$E[R(n)/S(n)] \sim dn^{0.5} \quad n \rightarrow \infty \quad (3.15)$$

d es una constante positiva, finita e independiente de n . La diferencia entre las ecuaciones 3.14 y 3.15 se conoce como *efecto Hurst* o *fenómeno de Hurst*.

3.6. Lento decaimiento de las varianzas

Estadísticamente, la característica más llamativa de los procesos auto-similares es la velocidad a la que cae la varianza muestral de los procesos $X^{(m)}$.

En los procesos cuyas series agregadas convergen en ruido blanco, la varianza muestral decae según m^{-1} (recíproco del nivel de agregación), mientras que en los procesos auto-similares esta velocidad es proporcional a $m^{-\beta}$ para $\beta \in (0, 1)$, (disminución más lenta).

Cox demostró que si la función de autocorrelación del proceso satisfacía la ecuación 3.10 (o equivalentemente, su función de densidad espectral se ajustaba a la ecuación 3.12),

$$\text{var}(X^{(m)}) \sim am^{-\beta} \quad m \rightarrow \infty \quad (3.16)$$

donde a es una constante positiva, finita e independiente de m , y $0 < \beta < 1$; de hecho el parámetro β es el mismo que en la ecuación 3.10 y está relacionado con el parámetro γ de la ecuación 3.12 por

$$\beta = 1 - \gamma$$

Por otra parte, para procesos estacionarios en covarianza cuyas series agregadas $X^{(m)}$ tienden a ruido blanco, es fácil comprobar que la secuencia $(\text{var}(X^{(m)}))$ satisface

$$\text{var}(X^{(m)}) \sim bm^{-1} \quad m \rightarrow \infty \quad (3.17)$$

donde b es una constante positiva, finita e independiente de m .

Las consecuencias del lento decaimiento de la secuencia $\text{var}(X^{(m)})$ en los procesos auto-similares pueden ser desastrosos para los tests estadísticos clásicos e intervalos de confianza y predicción. El motivo es que los errores estándar (derivados de los modelos convencionales) son de un factor que tiende a infinito a medida que aumenta el nivel de agregación de la muestra.

3.7. Análisis estadístico de las medidas de tráfico Ethernet

En este apartado, se va a establecer de una manera estadísticamente rigurosa la naturaleza auto-similar del tráfico Ethernet. Para ello, se van a describir las herramientas gráficas y estadísticas y se va a ilustrar su uso en el conjunto de medidas de tráfico coleccionadas en la red del departamento de Ingeniería Telemática.

3.7.1. Métodos estadísticos para comprobar la auto-similitud

A partir de las manifestaciones del carácter (exactamente o asintóticamente) auto-similar del conjunto de datos, se puede estimar el grado de auto-similitud (dependencia a largo plazo), parámetro H :

1. Dependencia a largo plazo \rightarrow análisis en el dominio de la frecuencia basado en el *periodogram*.
2. Lenta velocidad de decaimiento de las varianzas de los procesos agregados $X^{(m)} \rightarrow$ análisis de las varianzas de tales procesos.
3. Efecto Hurst \rightarrow análisis en el dominio del tiempo basado en el estadístico R/S.

El objetivo de estos análisis es inferir el grado de variabilidad (de auto-similitud) - parámetro H en la relación (3.14)) - del proceso auto-similar que presumiblemente ha generado las observaciones de la muestra bajo consideración.

Variance-time plots

El algoritmo de este análisis es el siguiente:

1. Se parte de una muestra de longitud n del proceso X : $\{X_1, X_2, \dots, X_n\}$.
2. Para cada $m : 2 \leq m \leq \frac{n}{2}$
 - a) Se divide la muestra de longitud n en k series de longitud m : $\{X_1, X_2, \dots, X_m\}$, $\{X_{m+1}, X_{m+2}, \dots, X_{2m}\}$, $\{X_{2m+1}, X_{2m+2}, \dots, X_{3m}\}$, \dots , $\{X_{(k-1)m+1}, X_{(k-1)m+2}, \dots, X_{km}\}$
 - b) Se calcula la media de cada una de esas series $X_1^{(m)}, X_2^{(m)} \dots X_k^{(m)}$ y la media total $\bar{X}^{(m)} = \frac{1}{k} \sum_{i=1}^k X_i^{(m)}$.
 - c) Se calcula la varianza de la media muestral como $Var[X^{(m)}] = \frac{1}{k-1} \sum_{i=1}^k (X_i^{(m)} - \bar{X}^{(m)})^2$.
3. Se representa $Var[X^{(m)}]$ frente a m en una escala logarítmica. La nube de puntos resultante se ajusta a una recta aplicando la técnica de mínimos cuadrados (los valores pequeños de m se ignoran para eliminar transitorios); la pendiente de esta recta se corresponde con el valor estimado de β ya que $Var[X^{(m)}] \sim cm^{-\beta}$ y por tanto

$$\log(Var[X^{(m)}]) \sim \log(c) - \beta \log(m)$$

Si $\hat{\beta} \in [-1, 0]$ se puede afirmar que el conjunto de datos exhibe un carácter auto-similar ($H \in [0, 5, 1]$).

$$H = 1 - \beta/2$$

Si la serie original es auto-similar, la varianza de los procesos agregados $X^{(m)}$ disminuye linealmente (se considera escala logarítmica) con el nivel de agregación; la pendiente de esta recta es arbitrariamente mayor que -1.

En los modelos dependientes a corto plazo (MMPP, modelos ARMA) la varianza de los procesos agregados tiene la forma (3.16); la recta que representa el decrecimiento de la varianza de $X^{(m)}$ con respecto a m , tiene una pendiente igual a -1.

La fiabilidad de esta herramienta gráfica depende del tamaño de la muestra. Para conjuntos de datos de tráfico Ethernet (número alto de observaciones en la muestra), "eyeball tests" como éste resultan de gran utilidad y dan una idea exacta de la naturaleza auto-similar de la serie de tiempo subyacente y del grado de auto-similitud.

Análisis R/S

Esta técnica es particularmente atractiva debido a su robustez frente a cambios en la distribución marginal (incluso distribuciones heavy-tailed). Si la distribución es gaussiana se pierde eficiencia y no se ha analizado la robustez del estadístico R/S.

- El conjunto de datos, la muestra de tamaño N ($X_k : k = 1, \dots, N$), se subdivide en k bloques sin solape de longitud $n/k > d$ ³.
- Se calcula el estadístico $R(t_i, d)/S(t_i, d)$ para cada uno de estos bloques de longitud d con comienzo en $t_i = [N/K](i - 1) + 1$, que satisfacen $(t_i - 1) + d \leq N$.

$$R(t_i, d) = \max(0, W(t_i, 1), \dots, W(t_i, d)) - \min(0, W(t_i, 1), \dots, W(t_i, d))$$

$$\text{donde } W(t_i, k) = \sum_{j=1}^k X_{t_i+j-1} - k\left(\frac{1}{d} \sum_{j=1}^d X_{t_i+j-1}\right), \quad k = 1, \dots, d.$$

$$S^2(t_i, d) \text{ es la varianza muestral de } X_{t_i}, \dots, X_{t_i+d-1}$$

La longitud de la muestra R/S resultante para cada valor de d decrece a medida que d aumenta, por la restricción, mencionada en el párrafo anterior, en los valores de t_i . $d_{t+1} = md_t$ con $m > 1$.

La representación gráfica en escala logarítmica de $R(t_i, d)/S(t_i, d)$ para cada valor de $\log(d)$ se denomina **pox diagram**. La nube de puntos⁴ se ajusta a una recta mediante la técnica de mínimos cuadrados. Las observaciones R/S correspondientes a valores pequeños de d se denominan correlaciones a corto plazo y las correspondientes a valores grandes de d son estadísticamente insignificantes si el número de observaciones por d es menor que 5. La pendiente de la recta de regresión es una estima del parámetro de Hurst. El número de bloques, k , y el número de valores d no debe ser demasiado pequeño.

Cuando el parámetro H de (3.13) está bien definido, la representación gráfica comienza con una zona transient manifestando la naturaleza de dependencia a corto plazo de la muestra, pero eventualmente se estabiliza

³Para cada valor de d se obtienen k valores de R/S , uno por bloque.

⁴no se consideran los valores R/S para valores altos de d

y fluctua en una línea recta de cierta pendiente.

Los análisis gráficos R/S se utilizan para determinar si tal comportamiento asintótico es apoyado por los datos. En caso afirmativo, una estima del parámetro de auto-similitud, \hat{H} , es dada por la pendiente asintótica de la recta (típicamente obtenida por aplicación de una técnica de ajuste de mínimos cuadrados).

En referencia a la efectividad del análisis R/S como función del tamaño de muestra, se pueden hacer comentarios similares a los del apartado anterior. Para propósitos prácticos, la característica más útil y atractiva de este análisis es su relativa robustez respecto a cambios en la distribución marginal. Esta característica permite investigaciones sobre la distinción de la propiedad de auto-similitud de una muestra y de sus características distribucionales.

Índice de variabilidad de las secuencias de paquetes de tráfico

Sea $N(t)$ el número de eventos (llegada de paquetes) de un proceso estacionario en el intervalo $(0, t]$. Para cada intervalo de tiempo fijo $\tau > 0$, se puede construir una secuencia de cuenta de eventos $Y = \{Y_n(\tau) > 0, n = 1, 2, \dots\}$, donde:

$$Y_n(\tau) = N[n\tau] - N[(n-1)\tau] \quad (3.18)$$

denota el número de eventos ocurridos durante el intervalo n de duración τ . Claramente, Y es también (débilmente) estacionario para todo $\tau > 0$.

En este estudio, Y representa una traza de tráfico de red donde $Y_n(\tau)$ denota el número de paquetes monitorizados desde un punto de la red durante el intervalo de tiempo n de duración τ , que es la escala de tiempo de la traza de tráfico, y representa la longitud (10 milisegundos, 1 segundo, 10 segundos, ...) de una muestra del proceso Y .

$$E\{N(t)\} = \frac{1}{E\{X\}} = \lambda t$$

donde $E\{X\}$ es la esperanza matemática de la variable aleatoria que representa el tiempo medio entre la llegada de dos mensajes consecutivos y λ es el valor medio de la tasa de llegadas.

El índice de dispersión de cuentas (IDC)⁵ se define como

$$IDC(t) = \frac{Var[N(t)]}{E\{N(t)\}} = \frac{Var[N(t)]}{\lambda t}$$

Una característica importante de IDC es que, matemáticamente, equivale al método *variance time plot* en la estimación del parámetro de Hurst de un proceso estocástico auto-similar.

⁵En un proceso Poisson IDC tiene valor 1

Si se representa en escala logarítmica $IDC(\tau)$ en el eje de ordenadas y τ en el eje de abscisas se obtiene una línea recta con pendiente $2H - 1$. Cuando Y es un proceso dependiente a largo plazo, la propiedad de decaimiento lento de la varianza es equivalente a una recta IDC con pendiente $1 - \beta < 0$. Cuando la curva IDC converge a una recta con pendiente 0 para algún valor finito de τ , Y es un proceso dependiente a corto plazo.

A continuación se va a definir una nueva medida de variabilidad:

Definición 1 *Para un proceso estacionario Y cuya función $IDC(\tau)$ es continua y diferenciable en el intervalo $(0, \infty)$, el **índice de variabilidad** en la escala τ es:*

$$H_v(\tau) = \frac{\frac{d(\log(IDC(\tau)))}{d(\log(\tau))} + 1}{2}$$

donde $\frac{d(\log(IDC(\tau)))}{d(\log(\tau))}$ es la pendiente de la recta que relaciona, en escala logarítmica, la función $IDC(\tau)$ y el nivel de agregación.

El índice de variabilidad se define para que en un proceso dependiente a largo plazo (o asintóticamente auto-similar), $H_v(\tau) = H \in (0,5, 1) \quad \forall \tau \geq \tau_0 > 0$. El valor de τ_0 depende del proceso particular. Si el proceso es exactamente auto-similar, $H_v(\tau) = H \in (0,5, 1) \quad \forall \tau > 0$; ésto es, si $\log(IDC(\tau))$ es lineal con respecto a $\log(\tau)$, $H_v(\tau)$ se reduce a H .

En general, el proceso Y exhibe una variabilidad significativa en aquellas escalas de tiempo τ tales que $0,5 < H_v(\tau) < 1$. Cuando $\frac{d(\log(IDC(\tau)))}{d(\log(\tau))} \rightarrow 1$, $H_v(\tau) \rightarrow 1$ implicando alta variabilidad.

Modelado de tráfico como multiplexación de conexiones de transporte

Algunas técnicas estadísticas y matemáticas describen y validan el fenómeno de escala presente en el tráfico agregado en términos de las propiedades más elementales de los patrones del tráfico generado por los usuarios individuales y/o aplicaciones. Ellos han impactado en la comprensión del tráfico de red actual hasta tal punto que, ahora se conoce por qué el tráfico de datos agregado exhibe un comportamiento auto-similar en un amplio rango de escalas temporales.

Una medida del éxito de esta nueva comprensión del tráfico de red es que los argumentos matemáticos son, al mismo tiempo, rigurosos y simples. Sin embargo, para proveer una descripción completa, es necesario comprender la naturaleza dinámica del tráfico en escalas pequeñas (milisegundos).

Debido al dominio de algunos protocolos y los mecanismos de control de congestión, que juegan un rol muy importante en las redes de datos y determinan el flujo de paquetes en las diferentes capas de la jerarquía de protocolos TCP/IP, el estudio de las características locales del tráfico de datos y la comprensión de las interacciones complejas (entre las diferentes conexiones y entre las capas de la jerarquía de protocolos en la red), están relacionados.

A continuación se va a demostrar por qué el fenómeno de escala (o auto-similitud) presente en el tráfico agregado de la red del departamento de Ingeniería Telemática, es una propiedad aditiva ¹.

4.1. Estructura aditiva y gaussianidad

El número de paquetes o bytes transmitidos por unidad de tiempo son originados por las conexiones activas durante el periodo de medida y cuya

¹Los procesos que generan el tráfico total exhiben alta variabilidad alta o características heavy-tailed

ruta atraviesa el enlace monitorizado.

En otras palabras, si para "escalas de tiempo " o "niveles de resolución" $m \gg 1$, $(X^{(m)} = X^{(m)}(k) : k \geq 0)$ denota el proceso de tasa de tráfico, esto es, número total de paquetes o bytes por unidad de tiempo (medido en escala de tiempo m) generados por todas las conexiones, se puede escribir:

$$X^{(m)} = \sum X_i^{(m)}(k) \quad k \geq 0 \quad (4.1)$$

donde se consideran todas las conexiones activas en el instante k y donde $X_i^{(m)}$ representa el número total de paquetes o bytes por unidad de tiempo (de nuevo medido en la escala de tiempo m) generados por la conexión i . Por tanto la ecuación 4.1 captura la naturaleza aditiva del tráfico de red agregado expresando el proceso de tasa de tráfico conjunto $X^{(m)}$ como una superposición de procesos de tasa de tráfico de las conexiones individuales.

Asumiendo por simplicidad, que los procesos de tasa de tráfico $X_i^{(m)}$ son independientes e idénticamente distribuidos (i.i.d), entonces bajo condiciones de débil regularidad sobre la distribución marginal de $X_i^{(m)}$ (existencia de segundos momentos), la ecuación 4.1 garantiza que el proceso de tasa de tráfico conjunto exhibe marginales gaussianas, siempre que el número de conexiones sea suficientemente alto.

4.1.1. Auto-similitud a través de conexiones *heavy-tailed*

Centrando la atención en la dinámica temporal de los procesos de tasa de tráfico individuales $X_i^{(m)}$, se supone por simplicidad que la conexión i envía paquetes o bytes a una tasa constante durante el periodo activo (*on period*) y no transmite información durante el periodo idle (*off period*).

En el entorno LAN considerado, una conexión corresponde a un par fuente-destino (*source-destination*) y los patrones de tráfico correspondientes conforman un proceso *alternating renewal*, donde los pares sucesivos de periodos *on* y *off* definen los intervalos *inter-renewal*.

En entornos WAN tales como Internet, se asocian conexiones individuales con **sesiones**. Una sesión comienza en un instante de tiempo aleatorio, genera paquetes (bytes) a una tasa constante durante el tiempo de vida de la conexión, y deja de transmitir; una sesión puede ser una aplicación ftp, una conexión telnet o una sesión web.

El interés de este trabajo se centra en las características globales de las conexiones:

- Patrón de llegadas.
- Duración de sesiones.
- Duración de los periodos *on/off*). El patrón de llegada de paquetes en el periodo activo ha sido modelado convenientemente asumiendo que la tasa de generación de paquetes es constante.

4.2. Variabilidad de la llegada de conexiones

En este apartado se va a analizar en un nivel intuitivo la auto-similitud del proceso de llegadas de conexiones TCP; su carácter intermitente sugiere la cuestión de cómo caracterizar la distribución del tiempo entre estos eventos.

4.2.1. Auto-similitud de la llegada de conexiones

Para estudiar el proceso de llegada de conexiones se establece una equivalencia entre los conceptos de **sesión** y **conexión** TCP para aplicaciones como telnet y ftp.

Por un lado, es cierto que los usuarios llegan al sistema de una forma más o menos sin memoria (*memoryless*), pero ésto no se traslada a la llegada de las conexiones TCP, porque una persona inicia más de una operación durante una sesión (independientemente de la clase de aplicación ftp, telnet o www). Ésto rompe el paradigma de Poisson y sugiere una nueva visión de la variabilidad del proceso de llegada de conexiones TCP.

Cuando las aplicaciones dominantes eran telnet y ftp-ctrl, el proceso de llegadas se ajustaba a un proceso Poisson (superposición de muchas fuentes cuya actividad era *memory-less*) - el hecho de que un usuario iniciara una sesión telnet no suponía un aumento de la probabilidad de iniciar otra sesión. Pero la aparición del www (*world wide web*) cambió este principio: una vez que un usuario empieza a navegar por internet la probabilidad de descargar otro conjunto de páginas web es mayor que la probabilidad de parar justo después de descargar la primera.

Entonces, el modelado estadístico más adecuado para el proceso de llegada de conexiones TCP se basa en procesos auto-similares, no procesos Poisson.

El grado de variabilidad de la llegada de conexiones se mantiene en todas las resoluciones de tiempo, la representación gráfica de los procesos agregados es muy "similar". Analíticamente se puede expresar con un único parámetro, parámetro de Hurst; cuanto más alto es este valor mayor es el grado de auto-similitud.

La representación gráfica del throughput (número de conexiones por unidad

de tiempo) en varias escalas de tiempo, exhibe semejanzas estructurales entre los procesos agregados.

4.2.2. Tiempo entre llegada de conexiones

A partir del proceso de llegada de conexiones TCP se deriva la distribución del tiempo transcurrido entre dos consecutivas.

La auto-similitud del proceso de llegada de conexiones TCP sugiere el uso de distribuciones estadísticas con cola pesada (*heavy-tailed*), tales como la distribución lognormal, Weibull y Pareto, para modelar adecuadamente el tiempo entre llegadas sobre un periodo de tiempo específico; no se va a emplear la distribución exponencial para caracterizar esta variable.

La función de distribución de probabilidad acumulada complementaria de una variable aleatoria *heavy-tailed* es:

$$P[X > x] \sim x^{-\alpha} \quad x \rightarrow \infty \quad (4.2)$$

donde $0 < \alpha < 2$. Una de las distribuciones *heavy-tailed* más simples es la distribución de Pareto, cuya función de densidad de probabilidad es dada por:

$$\rho(x) = \alpha k^\alpha x^{-\alpha-1} \quad (4.3)$$

donde $\alpha, k > 0$ y $x \geq k$. Su función de distribución tiene la forma

$$F(X) = P[X \leq x] = 1 - (k/x)^\alpha \quad (4.4)$$

El parámetro k representa el valor más pequeño de la variable aleatoria. Las distribuciones *heavy-tailed* tienen algunas propiedades que son cualitativamente diferentes de las distribuciones comúnmente encontradas tales como la distribución normal o exponencial.

Si $\alpha \leq 1$ el momento de primer orden de la distribución estadística es infinito y si $\alpha \in [1, 2]$, la varianza es infinita - a medida que α disminuye, una gran proporción de la masa de probabilidad se va desplazando hacia la cola. En términos prácticos, una variable aleatoria distribuida según una función potencial puede dar valores extremadamente grandes con una probabilidad no despreciable.

El efecto de modelar el tiempo idle (tiempo entre llegada de conexiones) según una distribución *heavy-tailed* es una variación positiva de la dependencia a largo plazo cuando α es próximo a 2, y una variación negativa a medida que el valor de α se acerca a 1.

4.3. Duración y tamaño de las conexiones

Considerando la limitación de los recursos de red y la interacción de los flujos de tráfico, la transferencia interactiva de ficheros cuyos tamaños se

distribuyen según una distribución *heavy-tailed* genera tráfico auto-similar. El grado de variabilidad del tráfico viene determinado por el parámetro α de la distribución estadística (distribución de Pareto). Para valores de α cercanos a 2, la representación del throughput se va aplanando en los niveles de agregación más altos, indicando una débil dependencia de la estructura en la serie subyacente. A medida que α se aproxima a 1, la variabilidad se preserva incluso a escalas mayores, indicando que la serie original posee dependencia a largo plazo.

Las series de tiempo agregadas generadas por una distribución **exponencial** y **Pareto** ($\alpha = 1,95$) son indistinguibles.

El tamaño de los ficheros transferidos en las conexiones TCP se distribuye según una variable aleatoria heavy-tailed (de cola pesada) debido a la naturaleza tan diversa de la información disponible en la red, desde pequeños ficheros de texto hasta vídeos de larga duración y alta calidad.

El valor de α de la función de distribución de probabilidad acumulada se calcula aplicando logaritmos en ambos lados de la igualdad,

$$F_c(x) \sim x^{-\alpha} \quad (4.5)$$

$$F_{jc} \sim l_j x^{-\alpha_j} L_j(x) \quad x \rightarrow \infty, 1 \leq \alpha_j \leq 2 \quad (4.6)$$

Se comprueba que la variabilidad de la duración de las conexiones es mayor que la del tamaño de los ficheros transferidos (valor de α menor); ésto es debido a la dinámica del protocolo TCP en presencia de congestión (aumento de la duración de las conexiones ante la detección de pérdidas de paquetes).

4.4. Tasa de transferencia de las conexiones

Modelo regular: la tasa de transmisión se asume constante e igual para todos los flujos (conexiones TCP).

La intermitencia del tráfico sintético obtenido a partir de este modelo es inferior a la ofrecida por el caso no regular.

Modelo no regular: la tasa de transferencia difiere para cada conexión TCP.

Este modelo se ajusta mejor a la traza real mediante una simple inspección visual.

Para realizar una comparación justa el valor de la tasa de transferencia para el caso regular debe ser tal que la carga en bytes coincida para ambos modelos.

En ambos modelos se asume que el proceso de llegadas es estacionario. Los modelos estacionarios que sean capaces de capturar las características del tráfico serán mucho más sencillos de tratar analíticamente que los modelos no estacionarios, por lo que son la primera solución a buscar; se deben escoger intervalos de tiempo en los que la hipótesis de estacionariedad sea razonable.

Las variables aleatorias que representan la duración de la conexión y el número de bytes transferidos tienen una cola de Pareto. En caso de ser independientes, su cociente (tasa de transferencia) debería tener también una cola de Pareto. Sin embargo la correlación de ambas variables da lugar a una distribución con un cola menos pesada (variable aleatoria Weibull)².

Matemáticamente, la propiedad *heavy-tailed* de, por ejemplo, el tiempo durante el que cada conexión individual genera paquetes de forma activa, implica que la correlación temporal de las versiones estacionarias de un proceso de tasa de tráfico individual $X_i^{(m)}$ y (debido a la propiedad de aditividad (4.1)) del proceso de tasa de tráfico conjunto, $X^{(m)}$, decae de forma hiperbólica (más rápido que el decaimiento exponencial); ésto es, exhiben dependencia a largo plazo.

Más precisamente, si $r^{(m)} = (r^{(m)} : k \geq 0)$ denota la función de correlación de la versión estacionaria del proceso de tasa de tráfico conjunto $X^{(m)}$, entonces la propiedad 4.5 implica dependencia a largo plazo.

Para todo $m \geq 1$, $r^{(m)}$ satisface:

$$r^{(m)}(k) \sim ck^{2H-2} \quad (4.7)$$

²Una característica destacable de la distribución de Weibull es que no tiene una cola tan pesada como para que su segundo momento no exista (aunque sí es más pesada que la de una variable aleatoria exponencial)

donde H mide el grado de dependencia a largo plazo en $X^{(m)}$.

Se puede expresar el parámetro H en términos del índice de cola $1 < \alpha < 2$:

$$H = (3 - \alpha)/2$$

Intuitivamente, la dependencia a largo plazo se manifiesta en periodos de tasas de tráfico mucho más altas y mucho más bajas que la media, independientemente de la escala de tiempo considerada. De hecho, para un proceso estacionario en covarianza y media nula, la ecuación 4.7 implica *auto-similitud asintótica de segundo orden*, ésto es, los procesos de tráfico conjunto $X^{(m)}$ tienen propiedades estadísticas de segundo orden idénticas y "parecen muy similares" para un número suficientemente alto de escalas temporales. En otras palabras, la ecuación (4.7) se mantiene si y solo si para escalas de tiempo suficientemente grandes m_1 y m_2 se cumple que:

$$m_1^{1-H} X^{(m_1)} \sim m_2^{1-H} X^{(m_2)} \quad (4.8)$$

donde H denota el parámetro de auto-similitud.

La capacidad de explicar la naturaleza auto-similar observada en el tráfico de datos agregado en términos de las propiedades estadísticas de las conexiones individuales que constituyen el proceso de tasa de tráfico conjunto, demuestra que este comportamiento asintóticamente auto-similar es:

- Una propiedad intrínsecamente aditiva (agregación de N conexiones).
- Es causado principalmente por las características del usuario/sesión/conexión (llegada de sesiones se distribuye según una variable aleatoria Poisson, la duración de las conexiones se ajusta a una distribución *heavy-tailed* con varianza infinita, ...).
- Es independientes de la red (de los protocolos dominantes y los mecanismos de control de congestión).

Para mantener la propiedad de auto-similitud en escalas de tiempo grandes únicamente es necesario que la distribución del número de paquetes o bytes transmitidos en una conexión sea *heavy-tailed* con varianza infinita. La naturaleza estadística de la transmisión de paquetes individuales en una conexión es irrelevante.

4.5. Efecto de la pila de protocolos

Se demuestra que los mecanismos empleados por el protocolo de la capa de transporte (control de congestión y fiabilidad de la comunicación) son factores importantes para mapear la distribución heavy-tailed del tamaño de los documentos transferidos a la auto-similitud del tráfico.

La auto-similitud del tráfico observado en el nivel de enlace y en el nivel de red es un fenómeno *inducido*, en parte influenciado por las acciones de la capa de transporte, permitiendo una evaluación intrínseca de la influencia y el impacto de los protocolos de transporte en el rendimiento de la red.

La transmisión fiable y los mecanismos de control de flujo de TCP sirven para preservar la estructura de dependencia a largo plazo, inducida por la distribución *heavy-tailed* del tamaño de los ficheros transferidos, y transferirla al siguiente nivel de la pila de protocolos (nivel de enlace). Ésto se manifiesta en un grado de variabilidad alto en todas las escalas de tiempo.

Por su parte, si el servicio ofrecido en el nivel de transporte de la jerarquía de protocolos no es fiable y no implementa ningún control de flujo (protocolo UDP), el tráfico exhibirá un carácter auto-similar en escalas de tiempo del orden de milisegundos.

El grado en el que el carácter *heavy-tailed* afecta a la auto-similitud se determina comprobando cómo el control de congestión es capaz de forjar una fuente de tráfico en un flujo de salida de media constante mientras se conserva la información.

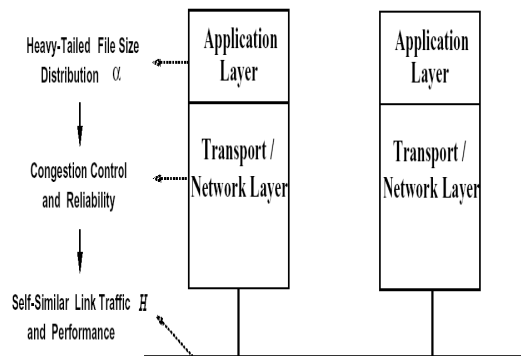


Figura 4.1: Influencia del nivel de aplicación en la auto-similitud del nivel de enlace.

4.6. Implicaciones de la auto-similitud en el rendimiento

En este apartado se va a evaluar el rol de la pila de protocolos, en particular, el protocolo de la capa de transporte para determinar el rendimiento de la red suponiendo un tráfico de entrada auto-similar. Para ello, se toma una visión dual de la pila de protocolos, dividiendo su funcionalidad en una capa de aplicación y una capa de red/transporte, agrupando las capas inferiores en el nivel de enlace. También se van a ignorar las subdivisiones más finas de las capas que hay por encima del nivel de transporte modelándolas como una simple entidad.

Los modelos de colas no son adecuados para predecir el rendimiento de una red con tráfico auto-similar. Con este tipo de tráfico el retardo aumenta y se requieren buffers de tamaño mayor. Norros desarrolló un modelo basado en el proceso FBM (fractional brownian motion) y un buffer de capacidad infinita con un tiempo de servicio constante.

El siguiente resultado de la capacidad requerida para el buffer, es representativo de las implicaciones del tráfico auto-similar para el rendimiento de la red.

$$q = \frac{\rho^{1/2(1-H)}}{(1-\rho)^{H(1-H)}}$$

donde H es el parámetro de Hurst. Para H=0.5 esta relación se simplifica a

$$q = \rho/(1-\rho)$$

que es la capacidad requerida para un buffer de un sistema con tiempos entre llegadas y tiempos de servicio que se distribuyen exponencialmente. Para un sistema con tiempo de servicio constante (M/D/1),

$$q = \frac{\rho}{(1-\rho)} - \frac{\rho^2}{2(1-\rho)}$$

Aunque el nivel de utilización de la red, ρ , no sea alto, es necesario un buffer de gran capacidad para tráfico auto-similar. De aquí se deriva una consecuencia obvia para el diseño del buffer: para altos niveles de utilización y presentando el tráfico un carácter auto-similar, los requerimientos de tamaño del buffer serán muy distintos de los análisis clásicos.

Algunos estudios analíticos y empíricos han demostrado que el tráfico auto-similar puede tener un impacto perjudicial en el rendimiento de la red:

1. Aumento del retardo.
2. Aumento de la tasa de pérdidas de paquetes.

Desde la perspectiva de la teoría de cola, una característica del tráfico dependiente a largo plazo es que la distribución de la longitud de la cola decae mucho más lentamente que para las fuentes de tráfico dependiente a corto plazo que poseen un decaimiento exponencial.

Mientras que estos resultados indican que la provisión de calidad de servicio (QoS) es un problema difícil en entornos donde el tráfico de datos es auto-similar, existen algunas vías para mitigar los efectos de la auto-similitud. En particular, una de las características del tráfico altamente auto-similar es que, aumentando el ancho de banda de la red mejora el retardo más que en el caso en el que el tráfico es débilmente auto-similar. De hecho, la reducción del retardo de paquetes a partir del ancho de banda aumentado es mucho más grande para tráfico auto-similar que para tráfico no auto-similar cuando la capacidad de los buffers no es demasiado pequeña. Ésto sugiere que, aunque reduciendo la capacidad del buffer se mejora el retardo en cola, una alternativa posiblemente más efectiva, es aumentar el ancho de banda.

En un entorno basado en UDP, el rendimiento (medido en términos de tasa de pérdida de paquetes y longitud media de cola) empeora drásticamente según aumenta el grado de auto-similitud.

Si se emplea una comunicación fiable vía TCP, la tasa de pérdida de paquetes (y de retransmisión) y el tiempo de transmisión de un fichero empeora suavemente a medida que aumenta el valor de H . La excepción es la longitud media de la cola, que presenta el mismo crecimiento superlineal que en el caso de la comunicación no fiable. El suave empeoramiento de las prestaciones de TCP bajo carga de tráfico auto-similar supone un incremento desproporcionado de la consumición de espacio en el buffer.

4.7. Modelado estocástico de la dependencia a largo plazo y las colas pesadas

Después de haber estudiado los principales conceptos de la dependencia a largo plazo y las distribuciones *heavy-tailed*, se consideran ahora los modelos propuestos para capturar estos fenómenos en el contexto del tráfico de red:

- Modelos de "caja negra": intentan imitar las tendencias observadas en los datos medidos sin capturar la naturaleza del tráfico.
- Modelos estructurales: intentan explicar las características de los datos observados empleando el conocimiento sobre el tráfico.

El primero de ellos es criticado en la literatura de ingeniería de red porque los parámetros de estos modelos no están relacionados con los parámetros de la red, y por tanto no son fáciles de interpretar; en ningún caso serán

útiles para propósitos de simulación.

A partir de una traza de tráfico se puede extraer gran cantidad de información, que se puede utilizar en el diseño de modelos estructurales. Por ejemplo, la cabecera de los paquetes TCP se puede utilizar para distinguir sesiones individuales o información del nivel de aplicación. Sin embargo, no es posible describir completamente la complejidad de los datos utilizando un modelo estructural, que significa que uno podría resort al modelado de "caja negra.

Debido al tamaño de las muestras que se manejan en el presente contexto de medidas Ethernet, es necesario un modelo sencillo debido al elevado número de parámetros necesarios para diseñar un modelo verdaderamente auto-similar.

Modelar, por ejemplo, la dependencia a largo plazo con la ayuda de procesos ARMA es equivalente a aproximar una función de autocorrelación que decae hiperbólicamente por una suma de exponenciales. Aunque siempre es posible, el número de parámetros necesarios tendería a infinito a medida que el tamaño de la muestra aumentara, y resultaría difícil dar una interpretación física de estos parámetros.

Capítulo 5

Naturaleza fractal del tráfico IT UC3M

En este capítulo se va a presentar un estudio estadístico de las trazas coleccionadas en el segmento de red del departamento de Ingeniería Telemática, que conecta aproximadamente 70 sistemas, incluyendo estaciones de trabajo UNIX y PCs.

5.1. Medidas de tráfico

El sistema de monitorización utilizado para coleccionar los datos del presente estudio registra todos los paquetes que circulan en el segmento de red bajo estudio con sellos de tiempo precisos.

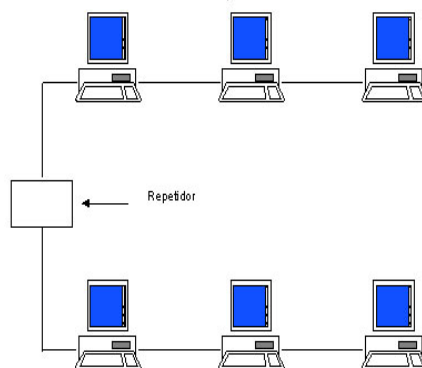


Figura 5.1: Escenario de la red de Ingeniería Telemática

Se puede establecer que los tiempos de observación usados para medir el tráfico en la red TCP/IP son lo suficientemente grandes para afirmar que el proceso estudiado corresponde a un proceso estacionario en sentido amplio (WSS) dentro de los tiempos de medición. También es importante tener en cuenta el concepto de carga de trabajo, es decir, la cantidad de trabajo repartido a un sistema por la unidad de llegada enésima, y que existen diferentes flujos de tráfico según los diferentes servicios que se proporcionen en la red.

Para contextualizar este apartado se va a resumir brevemente el conjunto de protocolos TCP/IP ¹.

Capa de enlace: incluye el driver del dispositivo con el sistema operativo y la interfaz de red correspondiente (de forma conjunta mantienen todos los detalles hardware de la interfaz física con el cable).

Capa de red: mantiene el movimiento de los paquetes en la red.

Capa de transporte: provee un flujo de datos entre dos hosts.

Capa de aplicación: mantiene los detalles de la aplicación particular.

Para satisfacer el objetivo de este capítulo (análisis estadístico del tráfico de la red de Ingeniería Telemática), se ha utilizado el analizador de protocolos **tcpdump** y un conjunto de filtros codificados en **awk**, que permiten una interpretación más cómoda de la información de las trazas de tráfico, cuyo formato es el siguiente:

| TIME | SOURCE | PORT | DESTINATION | PORT | FLAG | SEQNUM |
|-------------|-------------|-----------|-------------|-------|------|----------------------|
| 19:52:7314 | 406.17.8.12 | 64826 | 723.65.19.6 | www | S | 4256930:1256930(0) |
| 19:52:7318 | 723.65.19.6 | www | 406.17.8.12 | 64826 | S | 768500:768500(0) |
| ack 4256931 | | | | | | |
| 19:52:7322 | 406.17.8.12 | 64826 | 723.65.19.6 | www | . | |
| ack 768501 | | win 17520 | | | | |
| 19:52:7382 | 406.17.8.12 | 64826 | 723.65.19.6 | www | P | 4293561:4257101(170) |
| ack 768501 | | | | | | |

¹Sistema de 4 capas con diferentes protocolos en cada una de ellas.

La captura de tráfico se realizó el 25 de Marzo en la máquina **lm011.lab.it.uc3m.es** (ubicada en el segmento de red bajo estudio) durante 15 horas consecutivas (desde las 15:30 h. a las 06:30 h. del día 26).

El primer paso es filtrar el instante de llegada de cada unidad de tráfico, ejecutando la orden `awk '{print $1}'` siendo `$1` el índice del campo de selección. Así, se puede contabilizar en número de paquetes que han sido transmitidos durante el periodo de monitorización.

En la siguiente tabla se presenta una clasificación de los mensajes transmitidos según el protocolo de nivel superior (por encima del nivel de enlace).

| <i>Protocolo</i> | <i>Núm. paquetes</i> | <i>Porcentaje paquetes</i> |
|------------------|----------------------|----------------------------|
| TCP | 170676 | 60.3602 |
| UDP | 517172 | 20.3397 |
| ARP | 163486 | 2.91345 |
| ICMP | 5476 | 1.33207 |
| OTROS | - | 15.05 |

Tabla 5.1: Estadísticos de la tasa de paquetes transmitidos en distintas escalas de tiempo.

En el tiempo de análisis se establecieron (y finalizaron) 423 conexiones TCP, enviándose desde las máquinas de este entorno LAN hacia el exterior (*downstream*) 49166456 bytes, y en sentido contrario (*upstream*), 684917.

El filtro que se escribe a continuación calcula, a partir del instante de llegada de cada unidad de tráfico, la tasa de paquetes registrados en cada intervalo de tiempo (su longitud varía según según la escala).

Conviene especificar que la variable "x" es un parámetro configurable y depende del nivel de agregación (escala de interés).

```
{
  if(variable == 0){

    i++
    variable=1
    time_ref=$1
    if(i==1)
      time[i] =$1
    else{
      if(i>1){
        if(time_ref - time[i-1] < 0){

          contador++
          i--

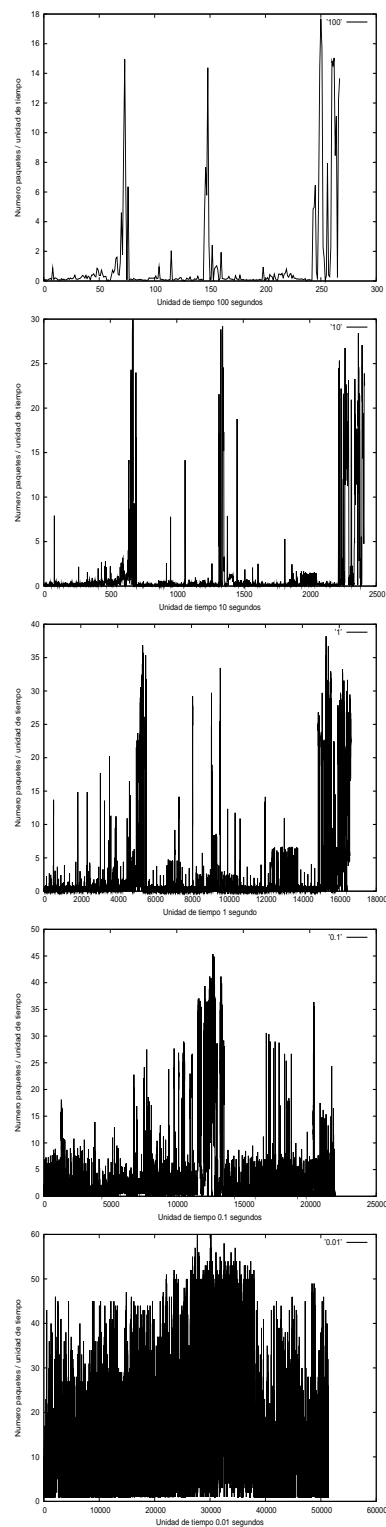
        }
        else{
          contador++
          time[i]=time_ref
        }
      }
    }
  }
  else{

    if (($1 - time_ref >= x) && ($1-time[i-1]>0)) {

      print time_ref,contador/(x*100)
      contador=0
      variable=0

    }
  }
}
```

En las siguientes gráficas se representa la tasa de paquetes monitorizados en diferentes escalas de tiempo (el estudio se realiza para diferentes niveles de agregación).



En la siguiente tabla se recogen algunos valores de interés en la caracterización del *throughput* (tasa de paquetes transmitidos).

| <i>Escala de tiempo</i> | $E[X]$ | $Var[X]$ | $Min[X]$ | $Max[X]$ |
|-------------------------|---------|----------|----------|----------|
| 0.01 segundos | 11.1909 | 125.237 | 9 | 1 |
| 0.1 segundos | 2.75609 | 7.59604 | 9.9 | 0.1 |
| 1 segundo | 1.70688 | 2.91345 | 9.93 | 0.01 |
| 10 segundos | 1.33023 | 1.76952 | 9.892 | 0.002 |
| 100 segundos | 1.15415 | 1.33207 | 8.9718 | 0.0042 |

Tabla 5.2: Estadísticos de la tasa de paquetes transmitidos en distintas escalas de tiempo.

La herramienta software **tcp-reduce** (cuyo código fuente se encuentra disponible en <http://ita.ee.lbl.gov/html/contrib/tcp-reduce.html>) ha sido empleada para recopilar, de forma resumida, la información de las conexiones TCP establecidas durante el periodo de monitorización. El conjunto de scripts que componen esta herramienta requieren un formato binario del fichero de entrada (añadir la opción -w en la ejecución del analizador de protocolos *tcpdump*). La información que proporciona la herramienta *tcp-reduce* consiste en:

Timestamp: instante de tiempo en el que se establece la conexión TCP.

Duración: tiempo transcurrido desde el establecimiento hasta la liberación de la conexión TCP.

Aplicación: protocolo cuya información se encapsula en el paquete transmitido

Bytes-US: número de bytes enviados hacia el exterior (internet).

Bytes-DS: número de bytes enviados hacia las máquinas de la red de Ingeniería Telemática .

Host-source: máquina que inicia el establecimiento de conexión.

Para recuperar cualquiera de estos datos en cada una de las conexiones basta con ejecutar el comando `awk '{print $x}'` sobre el fichero que contiene el resumen de la información, siendo x el índice de la selección.

Para calcular la tasa de transferencia de las conexiones en sentidos ascendente y descendente se ejecuta el siguiente filtro awk:

```
{
  #$4 si el sentido es ascendente y $5 si el sentido es descendente
  if($2 != "?" && $4 != "?")
    print ($4/$2)
}
```

El filtro que se escribe a continuación, toma como entrada un fichero con el instante de llegada de cada unidad de tráfico y calcula el tiempo que transcurre entre la llegada de dos paquetes (o conexiones TCP) consecutivos.

```
{
  if(NR==1){
    time_ref=$1
  }

  else{

    resultado=$1-time_ref

    if(resultado > 0){

      i++
      array[i]=resultado
      print $1-time_ref
      total=total+array[i];
    }
    time_ref=$1
  }
}
```

| <i>Variable</i> | $E[X]$ | $Var[X]$ | $Min[X]$ | $Max[X]$ |
|---------------------------------|------------------|-----------------|-------------|-------------|
| Duración conexiones | 76.9339 | 5918.83 | 99.0236 | 0.000102043 |
| Bytes upstream | 1201.52 | 1.44365e+06 | 10 | 985 |
| Bytes downstream | 17475.9 | 3.05407e+08 | 4 | 44429407 |
| Tasa transferencia (downstream) | 93799.7 | 8.7983+09 | 0.00327915 | 834824 |
| Tasa transferencia (upstream) | 2720.13 | 7.3991e+06 | 0.00357834 | 382267 |
| Tiempo entre conexiones | 7.51544 | 56.4819 | 0.000126839 | 9.99628 |
| Tiempo entre paquetes | $8,18673e^{-05}$ | $6,7022e^{-05}$ | 5.00679e-06 | 0.00019598 |

Tabla 5.3: Estadísticos generales para las conexiones TCP

Para calcular la media y varianza de las variables citadas anteriormente (duración de las conexiones, número de bytes transmitidos, tiempo entre paquetes, tiempo entre conexiones, throughput, ...) se emplea el siguiente filtro awk.

```

{
i++
a[i]
total+=$1
}

END{
    mean=total/NR
    print("Media")
    print(mean)

    for (k=0; k<i; k++){

        var=a[k+1]-mean
        var=var*var
        v=v+var

    }
    print("Varianza")
    print(v/NR)
}

```

| <i>Aplicación</i> | <i>Número bytes US</i> | <i>Porcentaje DS</i> | <i>Número bytes DS</i> | <i>Porcentaje US</i> |
|-------------------|------------------------|----------------------|------------------------|----------------------|
| www | 247417 | 11.61 | 199820 | 0.40 |
| imap | 546663 | 1.98 | 5375 | 0.01 |
| smtp | 6628 | 0.096 | 18401 | 0.037 |
| ftp | 4200 | 0.061 | 612 | 0.0012 |
| pop | 27267 | 0.398 | 2368 | 0.0048 |
| telnet | 122 | 0.0017 | 76 | 1.54e-04 |
| ssh | 6016177 | 87.9 | 48939597 | 17.02 |
| auth | 743 | 0.010 | 207 | 4.2e-04 |
| total | 6849217 | 100 | 49166456 | 100 |

Tabla 5.4: Bytes (*DS* y *US*) transmitidos durante el periodo de monitorización

| <i>Aplicación</i> | <i>Número conexiones</i> | <i>Porcentaje conexiones</i> |
|-------------------|--------------------------|------------------------------|
| www | 179 | 42.31 |
| imap | 27 | 6.38 |
| smtp | 24 | 5.67 |
| ftp | 24 | 5.67 |
| pop | 66 | 15.6 |
| telnet | 8 | 1.89 |
| ssh | 72 | 17.02 |
| auth | 23 | 5.43 |
| total | 423 | 100 |

Tabla 5.5: Conexiones TCP establecidas en el periodo de monitorización

El elevado porcentaje de conexiones del servicio *ssh* se debe al hecho de que la monitorización cubre el periodo nocturno, horas en las que se realiza una copia de seguridad de la información de las máquinas del segmento de red analizado.

| <i>Aplicación</i> | <i>Descripción</i> |
|-------------------|--|
| Imap | paquetes generados por el protocolo <i>simple mail transfer protocol</i> . |
| Smtip | paquetes generados por el protocolo <i>simple mail transfer protocol</i> . |
| Ftp | paquetes generados por el protocolo <i>file transfer protocol</i> (conexiones ftp-control y conexiones ftp-data). |
| Pop | paquetes generados por el protocolo <i>post office protocolo</i> . |
| Telnet | paquetes generados por el protocolo <i>remote terminal protocol</i> . |
| Ssh | paquetes generados por el protocolo <i>remote terminal protocol</i> . |
| Auth | paquetes generados por el protocolo <i>remote terminal protocol</i> . |
| Proto | paquetes generados por una colección de protocolos , incluyendo el protocolo <i>domain name service</i> y el protocolo <i>echo</i> . |

Tabla 5.6: Protocolos monitorizados

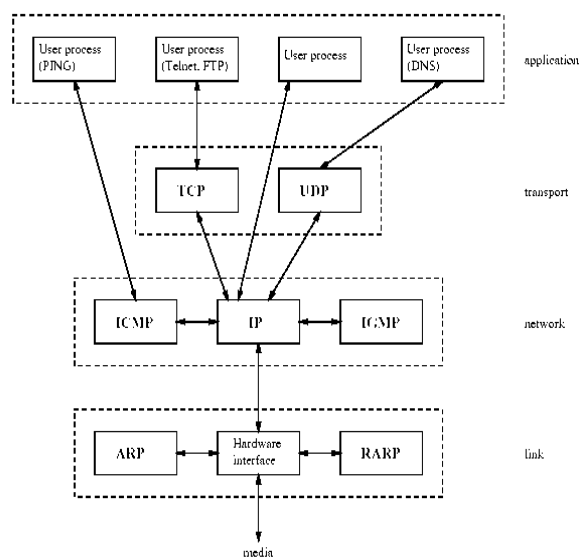


Figura 5.2: Jerarquía de protocolos TCP/IP

A continuación se representa gráficamente la función de distribución de probabilidad acumulada, $F(x)$ y su versión complementaria $F_c(x)$, de los parámetros que intervienen en los modelos de teletráfico.

Se puede comprobar que para el tiempo entre llegadas y la duración de las conexiones, la función de distribución de probabilidad acumulada corresponde a la de una variable aleatoria *heavy-tailed*, en la que el valor de α pertenece al intervalo [1.157 - 1.43] (varianza infinita). Sin embargo, para caracterizar la tasa de transferencia (en cualquiera de los dos sentidos), se debe emplear la distribución estadística Weibull (cola menos pesada que la de la distribución de Pareto).

Con el objetivo de implementar la función $F(x)$ se codifica el siguiente filtro awk. Para obtener la versión complementaria, $F_c(X)$, se ejecuta el comando `sort - r`.

```
{
    i++
    if(NR ==1) {
        mayor=$1
        menor=$1
    }
    else{
        if($1<menor)
            menor =$1

        if($1>mayor)
            mayor =$1
    }
    array[i]=$1
}END{
    intervalo =0.01

    variable = menor;
    print 0
    while (variable <= mayor){

        for (contador=1; contador<=i; contador++){
            if(array[contador]<=variable){
                cuenta++
            }
        }
        print cuenta/NR

        cuenta=0
        variable=variable+intervalo
        if(variable > mayor)
            variable=mayor
    }#fin del while
}
```

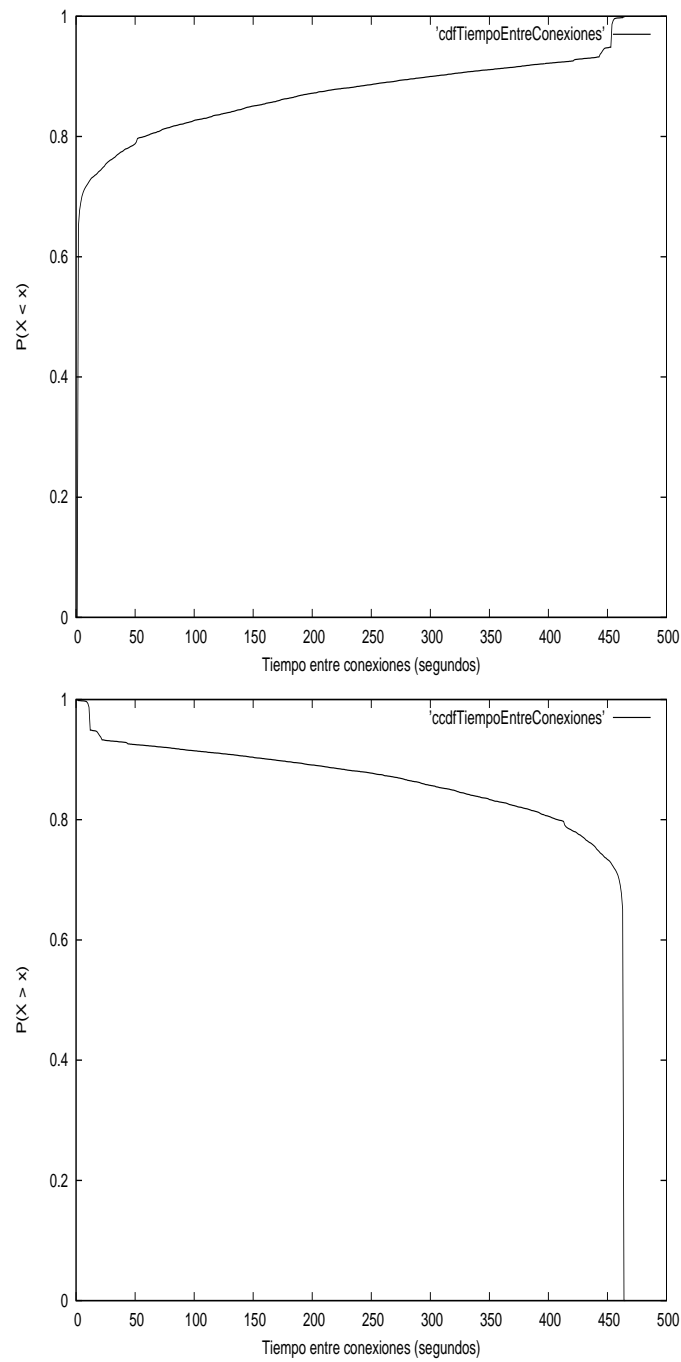



Figura 5.3: cdf (*cumulative distribution function*) y ccdf (*complementary cumulative distribution function*) del tiempo entre conexiones TCP.

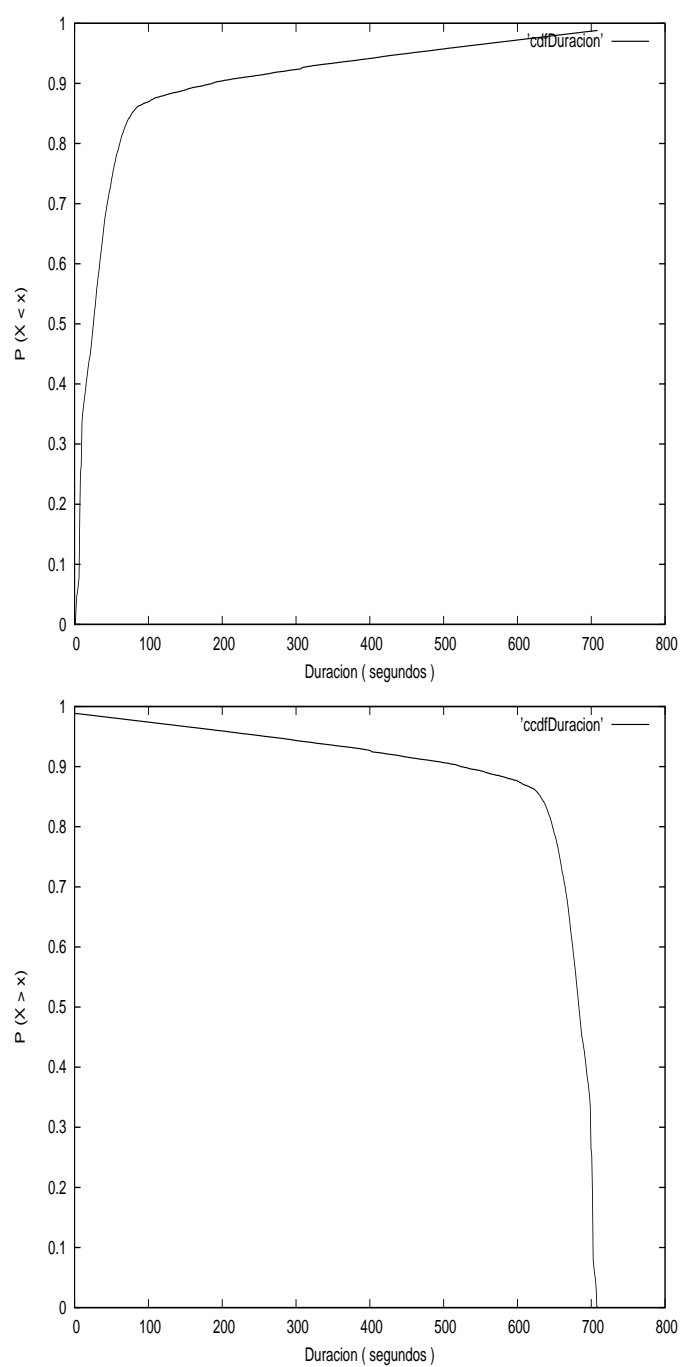


Figura 5.4: cdf (*cumulative distribution function*) y ccdf (*complementary cumulative distribution function*) de la duración de las conexiones TCP.

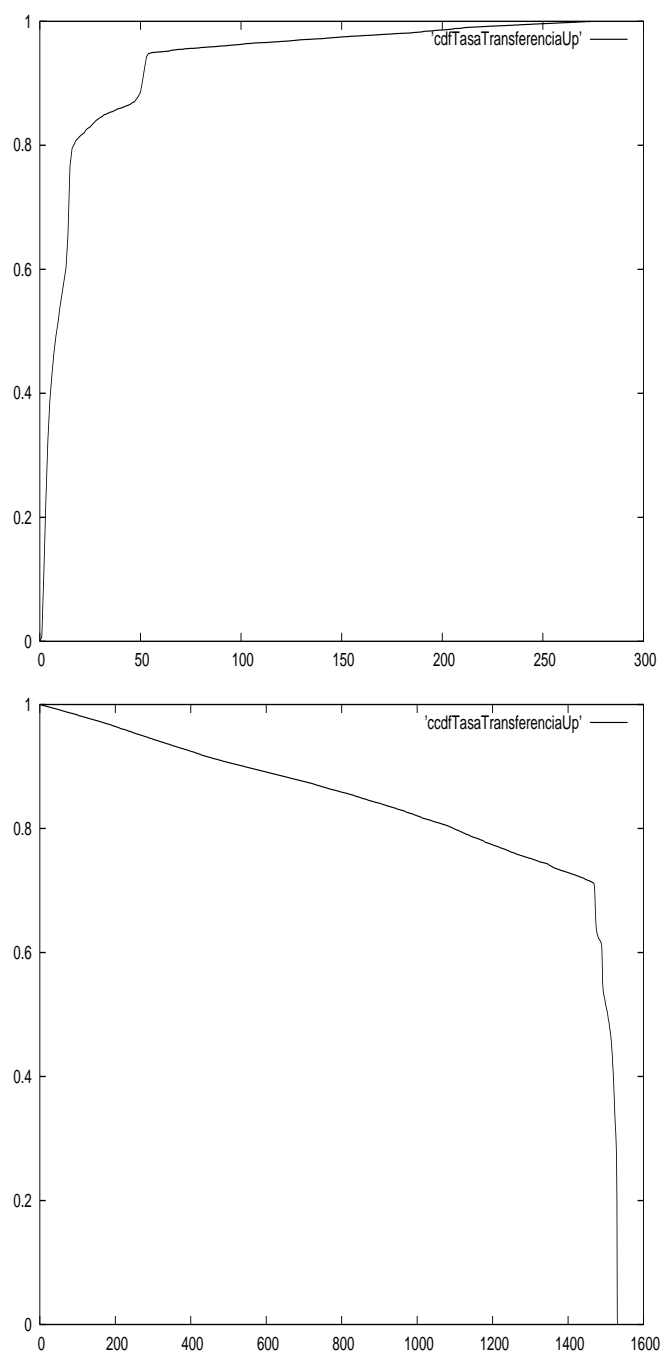


Figura 5.5: cdf (*cumulative distribution function*) y ccdf (*complementary cumulative distribution function*) de la tasa de transferencia en sentido ascendente.

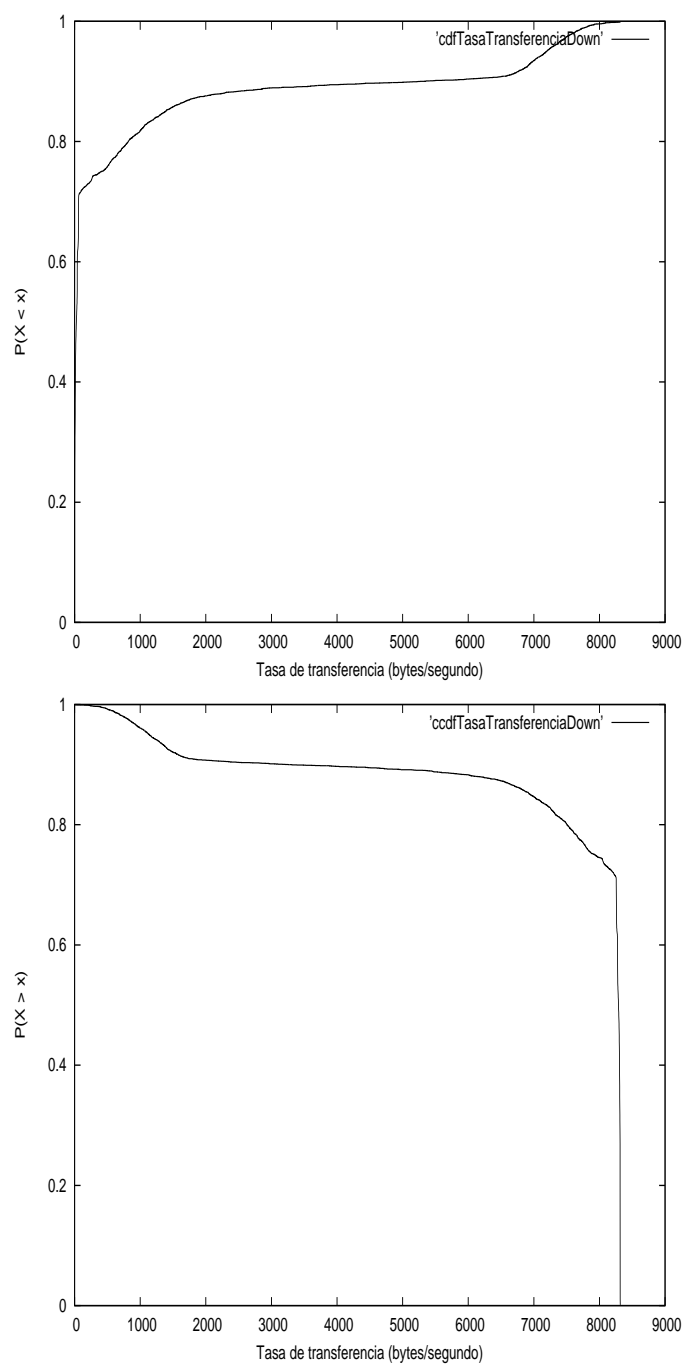


Figura 5.6: cdf (*cumulative distribution function*) y ccdf (*complementary cumulative distribution function*) de la tasa de transferencia en sentido descendente.

5.2. Evaluación de los estimadores sobre las trazas de tráfico

En las siguientes figuras se presenta la representación gráfica de tres estimadores del parámetro de Hurst en cada una de las escalas de interés en este estudio:

1. Estimador VT (*variance-time plot*).
2. Estimador RS (*rescaled adjusted range*).
3. Estimador IDC (*index dispersion counts*).

Para llevar a cabo este proceso de estimación se han utilizado tres programas codificados en C (disponibles en `ftp://ftp.ics.uci.edu/pub/duke/self-similar/`), que implementan los algoritmos descritos en el capítulo 3.

5.2.1. *Variance-time plot*

La ejecución del programa *variance.c* requiere los siguientes parámetros:

- Fichero de entrada: se debe especificar el sello de tiempo y la tasa de paquetes registrados en el intervalo k-ésimo (su longitud depende de la escala de observación).
- Unidad de tiempo: es el intervalo de medida por muestra (1 segundo).
- Logaritmo decimal del primer nivel de agregación considerado (típicamente es 0).
- Logaritmo decimal del último nivel de agregación considerado.
- Número de puntos en la gráfica entre las escalas logarítmicas.
Con esta cantidad y los dos parámetros anteriores, quedan definidos los límites de los niveles de agregación (en escala logarítmica) ².

El resultado de la estimación se guarda en un fichero (igual nombre que el fichero de entrada y extensión *vt*), cuyo formato es:

$$i \quad V$$

siendo *i* el índice de agregación y *V* la varianza muestral.

²Se calcula la varianza muestral de los procesos asociados a estos niveles de agregación.

5.2.2. *R/S plot*

El programa que implementa este algoritmo, *rslog.c*, requiere los mismos parámetros que el programa anterior.

Durante el proceso de estimación se crea un fichero temporal (igual nombre que el fichero de entrada y extensión *xj*) cuyo formato es:

$$n \quad \bar{X}(n) \quad \bar{Var}[X(n)]$$

El fichero de salida (igual nombre que el fichero de entrada y extensión *rs*) almacena el valor del estadístico R/S de los procesos asociados a cada nivel de agregación.

5.2.3. *Index dispersion counts*

La ejecución de este programa *idc.c* requiere los siguientes parámetros:

- Fichero de entrada: se debe especificar el sello de tiempo y la tasa de paquetes registrados en el intervalo *k*-ésimo (su longitud depende de la escala de observación).
- Logaritmo decimal del primer intervalo *L* (típicamente es 0).
- Logaritmo decimal del último intervalo *L* considerado.
- Número de puntos en la gráfica entre las escalas logarítmicas.
Con esta cantidad y los dos parámetros anteriores, quedan definidos la longitud de cada intervalo *L* (en escala logarítmica).

El fichero de salida (igual nombre que el fichero de entrada y extensión *idc*) almacena el valor de la función IDC para cada valor de *L*. Las figuras que se presentan a continuación son la representación gráfica (en escala logarítmica) de las relaciones que permiten estimar el valor del parámetro de Hurst \hat{H} :

- $\log Var(X^{(m)}) \sim -\beta \log(m)$
- $\log R/S(d) \sim H \log(d)$
- $\log IDC(m) \sim 2H - 1 \log(m)$

siendo *m*, *d* el nivel de agregación considerado.

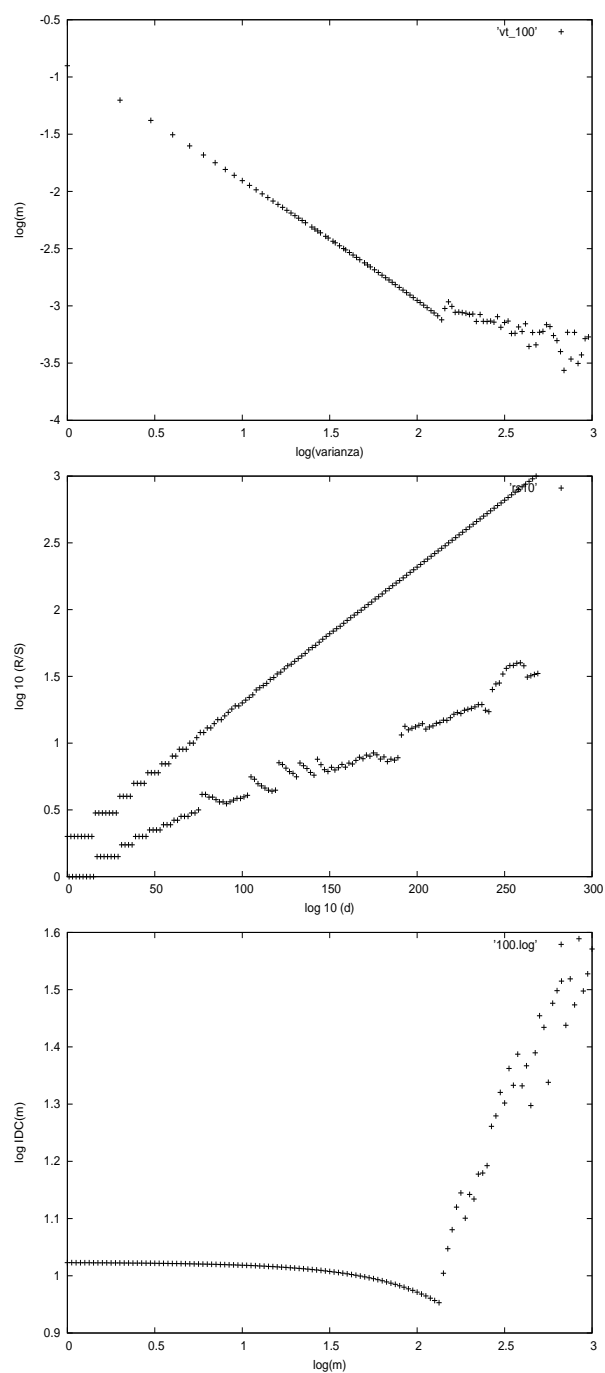


Figura 5.7: Nivel de agregación 100 segundos

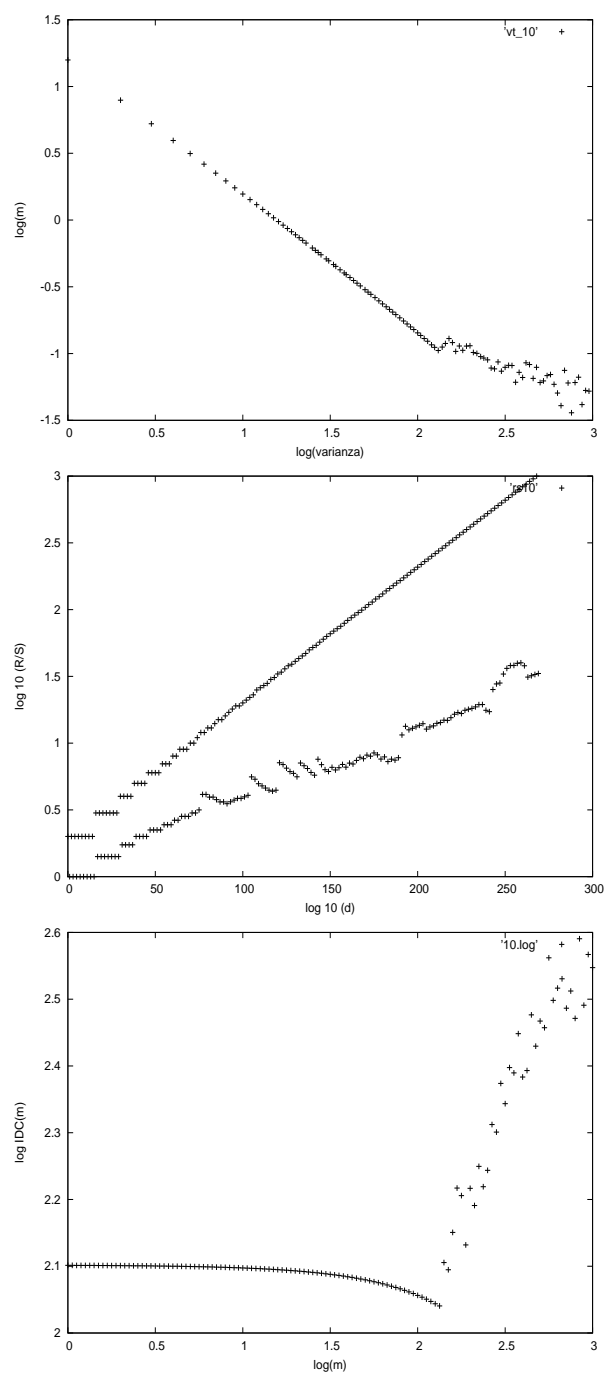


Figura 5.8: Nivel de agregación 10 segundos

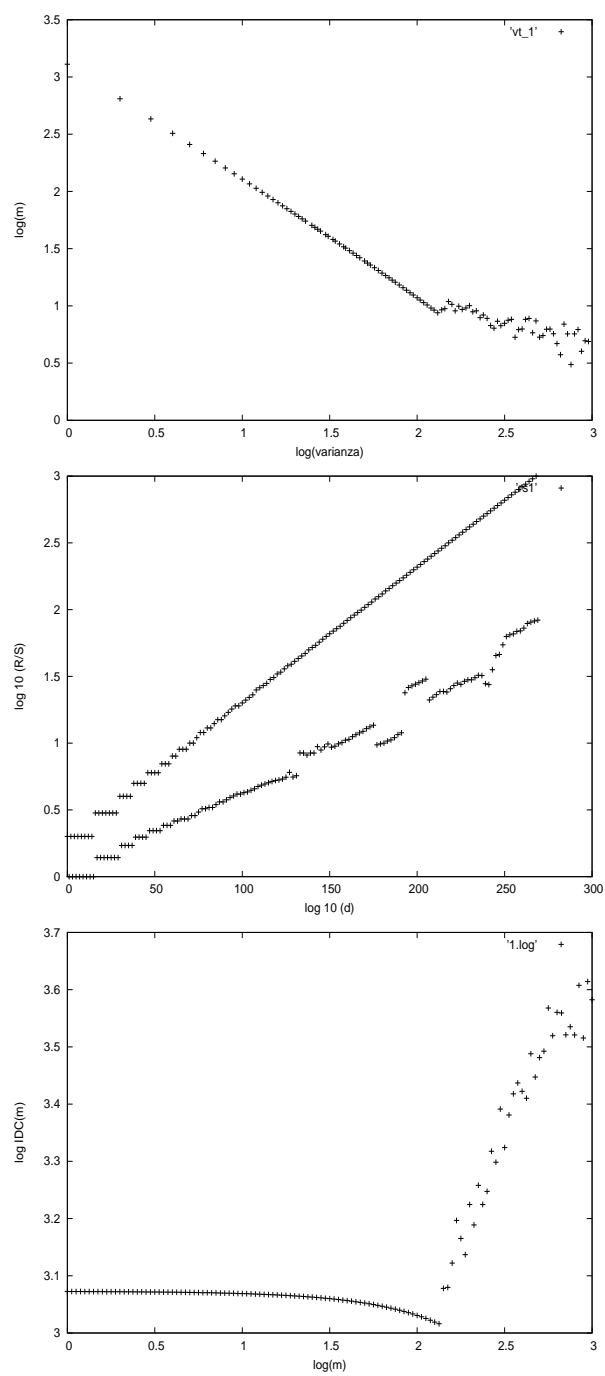


Figura 5.9: Nivel de agregación 1 segundo

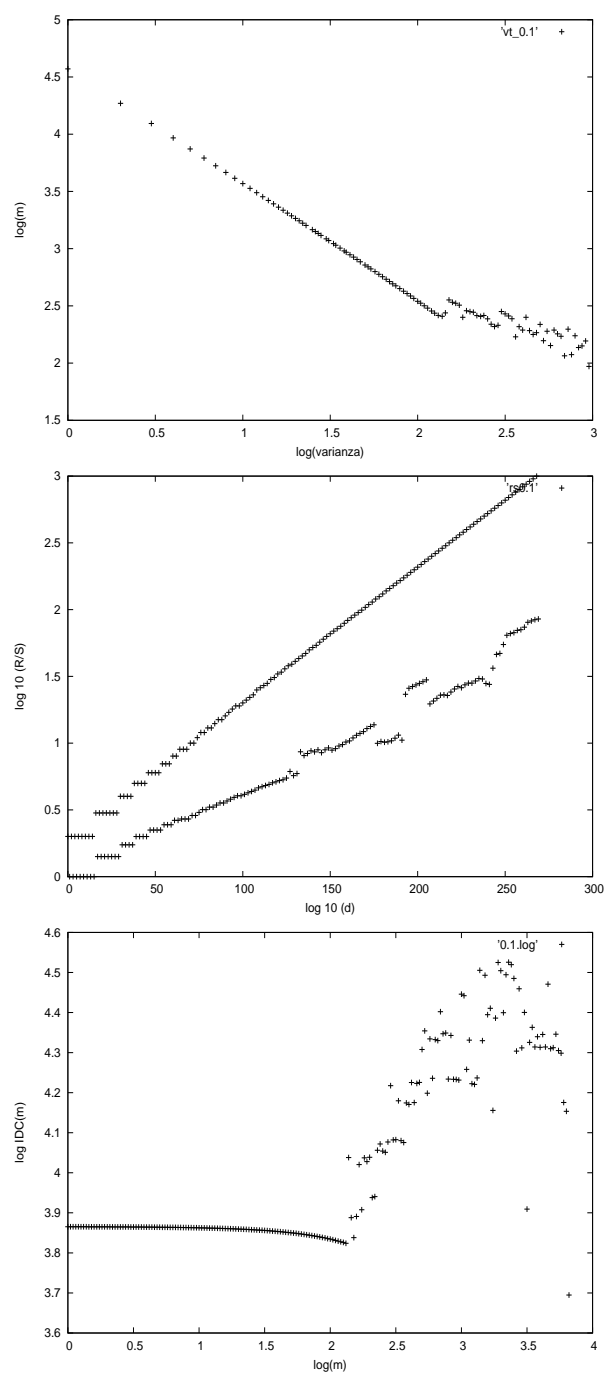


Figura 5.10: Nivel de agregación 0.1 segundos

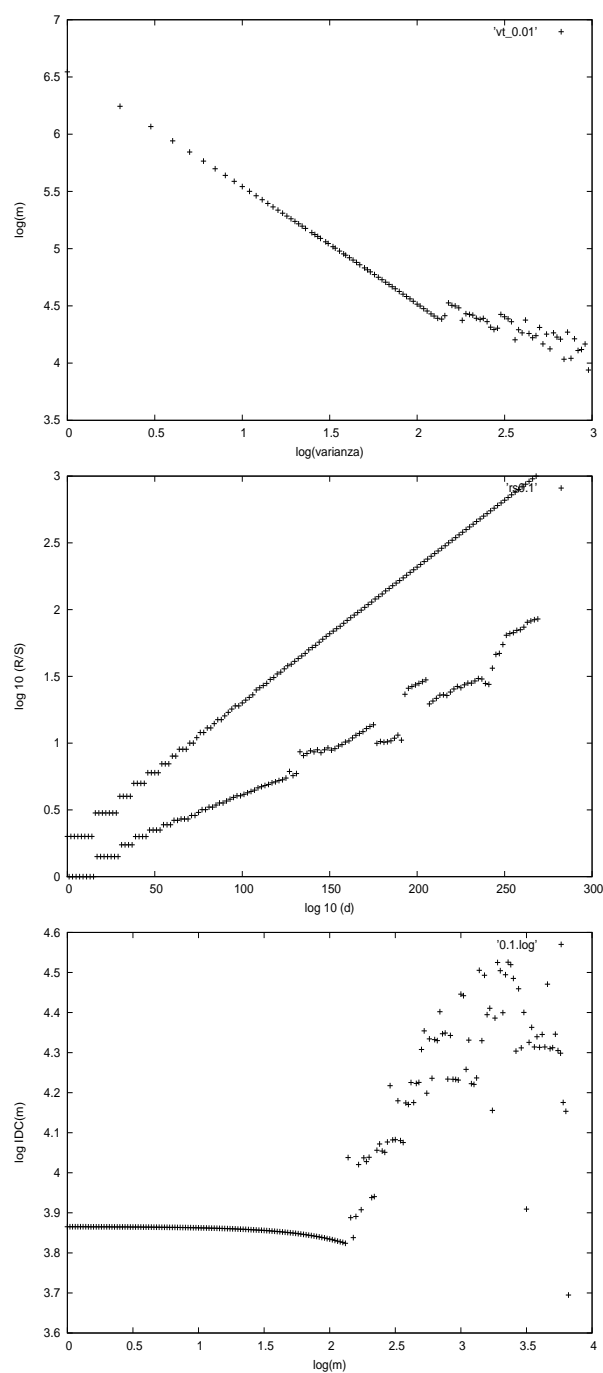


Figura 5.11: Nivel de agregación 0.01 segundos

La nube de puntos que resulta después de aplicar cada uno de los estimadores, es aproximada por una recta mediante la técnica de mínimos cuadrados. El valor de la pendiente de estas rectas se presenta en la tabla

| Escala de tiempo | Pte recta VT | Pte recta RS | Pte recta IDC |
|------------------|--------------|--------------|---------------|
| 100 segundos | 0.1341 | 0.487 | 0.118 |
| 10 segundos | -0.8184 | 0.527 | 0.1206 |
| 1 segundos | -0.7902 | 0.613 | 0.1384 |
| 0.1 segundos | -0.7755 | 0.6568 | 0.163 |
| 0.01 segundos | -0.7759 | 0.6537 | 0.1694 |

Tabla 5.7: Valores de la pendiente de la recta que aproxima (mediante la técnica de mínimos cuadrados) la nube de puntos de la estimación

5.8.

| Escala de tiempo | \hat{H}_{VT} | \hat{H}_{RS} | \hat{H}_{IDC} |
|------------------|----------------|----------------|-----------------|
| 100 segundos | 0.5936 | 0.487 | 0.559 |
| 10 segundos | 0.5908 | 0.527 | 0.5603 |
| 1 segundos | 0.6049 | 0.6613 | 0.5692 |
| 0.1 segundos | 0.6122 | 0.6568 | 0.5815 |
| 0.01 segundos | 0.6120 | 0.6537 | 0.5847 |

Tabla 5.8: Valores estimados de H (\hat{H})

Generación de tráfico fractal

Los modelos de tráfico fractal presentan serias dificultades a la hora de resolver de forma analítica problemas de dimensionamiento de enlaces, tamaños de buffers, técnicas de control de admisión, ... Generalmente estos modelos permiten poner de manifiesto factores que pueden ser la causa de las diferentes características de los procesos de tráfico reales. Ante la dificultad de resolver analíticamente los sistema de interés se recurrirá a trazas de tráfico generadas en base al modelo, con las mismas características principales que el tráfico real, que permitan poner a prueba los sistemas o realizar extensivos estudios numéricos.

En este capítulo se van a presentar varias técnicas existentes de generación de tráfico según un proceso FGN.

1. Considerar un proceso $R(t)$ en el que la duración de los periodos *on/off* siga una distribución de cola pesada (Pareto)
El proceso $S_n(t)$ se obtiene después de multiplexar n instantes independientes del proceso $R(t)$, donde $S_n(t)$ es el número de procesos $R(t)$ que están en el periodo ON en el instante t . La serie temporal $S_n(t)$ es asintóticamente auto-similar.
2. Considerar un modelo de colas $M/G/\infty$
La llegada de unidades de tráfico sigue una distribución de Poisson y el tiempo de servicio sigue una distribución de cola Pesada con varianza infinita. En este modelo X_t es el número de ítems en el sistema en cada instante t ; X_t es asintóticamente auto-similar.
3. "Random Midpoint displacement" (RMD)
Este método subdivide progresivamente un intervalo. En cada división se utiliza un desplazamiento gaussiano para determinar el valor de la observación en la mitad del subintervalo.
La auto-similitud viene por la apropiada escala de la varianza del desplazamiento. Este método resulta atractivo porque es rápido y se puede utilizar para interpolar una traza auto-similar entre observaciones hechas en una escala de tiempo grande.

4. Calcular coeficientes de onda correspondientes a una transformada de ondas de FBM

Los coeficientes son usados con una transformación inversa de ondas para normalizar la traza de FBM. Este método es sólo aproximado porque los coeficientes de onda no son independientes pero es difícil capturar su interdependencia.

5. Utilizar una aproximación a la definición de FBM

En vez de calcular para cada nuevo punto la contribución correlacional de los puntos previos (definición de procesos FBM), se computa sólo la contribución agregada. La aproximación puede producir n puntos en un tiempo $O(n)$ en vez de $O(n^2)$.

La mayoría de los métodos de síntesis requieren mucho tiempo de CPU, a excepción del método RMD, que es bastante rápido.

6.1. Algoritmo FFT

6.1.1. Estimador Whittle

Un problema clave cuando se estudian las muestras de los procesos auto-similares es estimar el parámetro de Hurst, parámetro H . A continuación se va a repasar el estimador de Whittle, porque algunas de las propiedades de los procesos FGN sobre el que se basa son también utilizadas por el método de síntesis FGN, y porque una aproximación clave utilizada por el método puede utilizarse también para evaluar rápidamente el estimador Whittle.

$\{x_t\}$ es una muestra de un proceso auto-similar X del que son conocidos todos los parámetros excepto la varianza y el parámetro H .

$f(\lambda, H)$ denota el espectro de potencia de X cuando se normaliza para tener varianza 1, e $I(\lambda)$ es el periodogram (espectro de potencia estimado a partir de la transformada de Fourier de $\{x_t\}$). Entonces para estimar H , se busca un \hat{H} que minimiza:

$$g(\hat{H}) = \int_{-\pi}^{\pi} \frac{I(\lambda)}{f(\lambda; \hat{H})} d\lambda \quad (6.1)$$

Si $\{x_t\}$ tiene longitud n , entonces la integral anterior se convierte en una suma discreta sobre las frecuencias:

$$\lambda = \frac{2\pi}{n}, \frac{4\pi}{n}, \dots, \frac{2(n-1)\pi}{n}$$

La forma de este estimador se debe a que el periodograma $I(\lambda)$ son asintóticamente independientes y exponencialmente distribuidas con media $f(\lambda; H)$.

Junto con el estimador se puede calcular σ_H^2 , su varianza:

$$\sigma_H^2 = 4\pi \left[\int_{-\pi}^{\pi} \left(\frac{\log f(\omega)}{H} \right)^2 d\omega \right]^{-1}$$

Cuando se sintetizan muestras de tráfico auto-similares, se puede utilizar el estimador Whittle junto con σ_H^2 para determinar si \hat{H} es una aproximación aceptable de H .

El estimador Whittle no es un test para comprobar si una muestra de una serie temporal es dependiente a largo plazo, es un estimador de H , asumiendo que el espectro de potencia del proceso subyacente corresponde a $f(\lambda; H)$.

6.1.2. Método FFT (*Fast Fourier Transform*)

En este apartado se presenta un método basado en la transformada de Fourier discreta en el tiempo (DTFT) para sintetizar muestras del proceso FGN (*Fractional Gaussian Noise*).

La implementación del método es tanto más rápida que las técnicas existentes y genera muestras que en la mayoría de los casos son indistinguibles de un proceso FGN real, utilizando técnicas estadísticas. En particular, el método FFT es menos parcial que el método RMD (el otro algoritmo más rápido que se va a estudiar).

La estrategia detrás de este método consiste en construir una secuencia de números complejos z_i correspondientes al espectro de potencia del proceso; $\{z_i\}$ es una muestra del proceso en el dominio de la frecuencia. Se puede utilizar una transformada de Fourier inversa para obtener $\{x_i\}$, contrapartida de $\{z_i\}$ en el dominio del tiempo.

La serie $\{x_i\}$ tiene (por construcción) el espectro de potencia del proceso FGN. Debido a que la función de autocorrelación y el espectro de potencia forman una pareja Fourier, la función de autocorrelación de la serie $\{x_i\}$ tiene las mismas características que la del proceso FGN.

La dificultad de esta aproximación radica en el cálculo exacto de $f(\lambda, H)$ y en encontrar la secuencia $\{z_i\}$ correspondiente al espectro de potencia FGN. No existe ninguna razón a priori para asumir que z_i son independientes, y capturar su interdependencia puede resultar difícil.

La transformada de Fourier discreta en el tiempo (DTFT) y su inversa pueden ser rápidamente calculadas utilizando el algoritmo *Fast Fourier Transform (FFT)*. Por tanto el método que se presenta es un método FFT de síntesis de ruido gaussiano fractal (muestras de un proceso FGN). Los resultados del método son FGN en sentido *estricto* pero sí en sentido *efectivo*, ésto es, son indistinguibles (utilizando los tests estadísticos actuales) de las muestras de un proceso estocástico FGN. Por tanto, para propósitos prácticos las muestras pueden considerarse FGN con un alto grado de confianza.

En línea con este argumento existen cuatro tests que una muestra del proceso FGN debe superar:

1. Una representación *variance-time plot* debería mostrar que, si la muestra es agregada por un factor de m , la varianza de la versión agregada decae con una velocidad de $m^{-\beta}$, donde $\beta = 2(1 - H)$. Éste es un test heurístico en el sentido de que las propiedades estadísticas de estas gráficas no se conocen, pero resulta de gran utilidad porque se evidencia el carácter intermitente de la muestra a distintas escalas de tiempo.
2. El test de bondad de ajuste de Beran para la dependencia a largo plazo debe indicar que la muestra es consistente con este fenómeno.
3. Estimador Whittle debe producir una estima de H , \hat{H} consistente con el verdadero valor de H utilizado para generar la muestra.
4. La distribución marginal de la muestra debe ser normal o muy cercana a la normal, ya que corresponde a un proceso gaussiano. Ésto puede ser comprobado utilizando el test *Anderson-Darling*.

El test de Beran y el estimador Whittle están relacionados de forma compleja con el espectro de potencia del proceso estimado. Para un proceso FGN, el espectro de potencia es:

$$f(\lambda; H) = A(\lambda; H)[|\lambda|^{-2H-1} + B(\lambda; H)] \quad (6.2)$$

para $0 < H < 1$ y $-\pi < \lambda < \pi$, donde

$$A(\lambda; H) = 2\sin(\pi H)\Gamma(2H + 1)(1 - \cos\lambda)$$

$$B(\lambda; H) = \sum_{j=1}^{\infty} [(2\pi j + \lambda)^{-2H-1} + (2\pi j - \lambda)^{-2H-1}] \quad (6.3)$$

La principal dificultad con el uso de la ecuacion 6.2 para calcular el espectro de potencia es la suma infinita en la expresión de $B(\lambda; H)$, que se va a aproximar como sigue a continuación:

$$B(\lambda; H) \sim a_1^d + b_1^d + a_2^d + b_2^d + a_3^d + b_3^d + \frac{a_3^{d'} + b_3^{d'} + a_4^{d'}}{8H\pi} \quad (6.4)$$

donde

$$d = -2H - 1$$

$$d' = -2H$$

$$a_k = 2k\pi + \lambda$$

$$b_k = 2k\pi - \lambda$$

Se define $f(\lambda; H)$ como la aproximación de la ecuación 6.2 obtenida utilizando la simplificación 6.3 para $B(\lambda; H)$. Las entradas del método FFT son H (parámetro Hurst deseado) y n (número par de observaciones de la muestra sintética). El método procede como sigue:

- Construir una secuencia de valores $\{f_1, \dots, f_{n/2}\}$, donde $f_j = f(\frac{2\pi j}{n}; H)$, correspondiente al espectro de potencia de un proceso FGN para frecuencias de $2\pi/n$ a π .

- Modificar cada $\{f_i\}$ multiplicándola por una variable aleatoria exponencial de media 1. Esta secuencia modificada es $\{\hat{f}_i\}$. Ésto se hace porque cuando se estima el espectro de potencia del proceso utilizando el periodograma de una muestra, la potencia estimada para cada frecuencia está distribuida asintóticamente como una variable aleatoria exponencial con media igual a la potencia actual.

Una cuestión referente a la precisión del método es el grado con el que este resultado asintótico puede ser aplicado a un espectro de potencia finito sin comprometer la propiedad de auto-similitud.

- Construir $\{z_1, \dots, z_{n/2}\}$, una secuencia de valores complejos tal que $|z_i| = \sqrt{\hat{f}_i}$ y la fase de z_i está uniformemente distribuida entre 0 y 2π . La técnica de aleatorización de la fase preserva el espectro de potencia (y por tanto la función de autocorrelación) correspondiente a $\{\hat{f}_i\}$, pero asegura que las muestras generadas utilizando el método serán independientes. Ésto también hace que la distribución marginal del resultado sea normal, un requisito para el proceso FGN, y también para aplicar el procedimiento Whittle utilizando una expresión para $f(\lambda; H)$ correspondiente al espectro de potencia FGN.

Una cuestión es por qué la aleatorización de la fase lleva a un proceso gaussiano (se verifica que la ausencia de aleatorización de la fase supone un proceso no gaussiano). Esta técnica hace que las componentes de las distintas frecuencias sean independientes, cuando los correspondientes senos y cosenos se suman en el cálculo de la operación de la transformada inversa de Fourier, el proceso se ajusta con una versión del teorema central del límite probado por Lindeberg. Este teorema establece que si se suman variables aleatorias independientes y distribuidas de forma diferente, la suma converge a una distribución normal, que es el objetivo.

La condición de Lindeberg requiere que las varianzas de las diferentes distribuciones sean pequeñas en comparación con la suma total. Si ésto se verifica, entonces el teorema explica por qué la técnica de aleatorización lleva a una distribución marginal normal.

- Construir $\{z'_0, \dots, z'_{n-1}\}$, una versión expandida de $\{z_1, \dots, z/n/2\}$:

$$z'_j = \begin{cases} 0 & \text{si } j = 0 \\ z_j & \text{si } 0 < j < n/2 \\ z_j/z_{n-j} & \text{si } n/2 < j < n \end{cases}$$

donde z_{n-j} denota el complejo conjugado de z_{n-j} .

$\{z'_j\}$ es el espectro de potencia utilizado para construir $\{z_i\}$ pero porque ésto es simétrico sobre $z'_{n/2}$ ésto ahora corresponde a la transformada de Fourier de una señal de un valor real.

- Transformada inversa de Fourier $\{z'_j\}$ para obtener la muestra FGN aproximada $\{x_i\}$.

El método FFT (*Fast Fourier Transform*) no es una respuesta final a la simulación de la auto-similitud en el tráfico de red, simplemente es un punto de partida.

6.2. Algoritmo RMD

El objetivo de este algoritmo (*random midpoint displacement*) es generar una traza FBM en el intervalo de tiempo $[0, T]$. Para ello se va a subdividir el intervalo $[0, T]$ recursivamente y se van a calcular los valores del proceso en el punto medio.

Se va a suponer un tráfico auto-similar en el intervalo $[0, T]$ con coeficiente de muestreo n , donde $T = 2^n$.

Pasos del algoritmo RMD:

1. Generar tráfico acumulado en los instantes $t = 0$ y $t = T$ para el intervalo de tiempo $[0, T]$.
2. El tráfico en el punto medio $t = \frac{T}{2}$ es generado de acuerdo con unos estadísticos (que se indicarán), obteniéndose dos subintervalos - $[0, \frac{T}{2}]$ y $[\frac{T}{2}, T]$.
3. Generar tráfico para los puntos $t = \frac{T}{4}$ y $t = \frac{3T}{4}$. Sobre este paso se van a realizar N iteraciones.

La regla para generar tráfico en el punto medio, $Z(\frac{a+b}{2})$, según el valor del tráfico en los puntos extremos, $Z(a)$ y $Z(b)$, se basa en la aproximación FBM. Específicamente, algunos autores han descubierto que el desplazamiento del punto medio, $Z_{RMD} = \frac{Z(\frac{a+b}{2}) - [Z(a) + Z(b)]}{2}$, es independiente de $Z(a)$ y $Z(b)$ y que se distribuye según una función gaussiana de media nula si $Z(t)$ es un proceso FBM.

El algoritmo RMD utiliza una secuencia de variables aleatorias gaussianas independientes y las toma con la escala adecuada como un desplazamiento para ajustar el tráfico en cada punto medio.

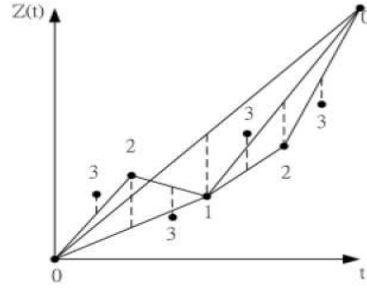


Figura 6.1: Ilustración de las tres primeras iteraciones del algoritmo RMD

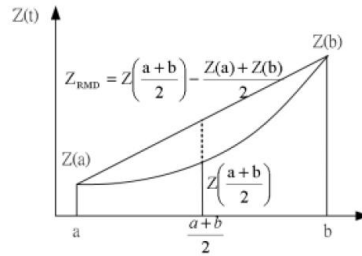


Figura 6.2: Relación entre $Z((a+b)/2)$, $Z(a)$, $Z(b)$ y Z_{RMD}

La varianza del desplazamiento del punto medio se determina a partir de la siguiente derivación.

$$\begin{aligned}
 Var[Z_{RMD}] &= Var\left[Z\left(\frac{a+b}{2}\right) - \frac{Z(a) + Z(b)}{2}\right] = \\
 &= \frac{1}{4} Var\left[\left(\frac{Z(a+b)}{2} - Z(a)\right) - \left(Z(b) - Z\left(\frac{a+b}{2}\right)\right)\right] = \\
 &= \frac{1}{4} (Var[Z(\frac{a+b}{2}) - Z(a)] + Var[Z(b) - Z(\frac{a+b}{2})] - 2Cov[Z(b) - Z(\frac{a+b}{2}), Z(\frac{a+b}{2}) - Z(a)]) = \\
 &= \frac{1}{4} \left[\left(\frac{b-a}{2}\right)^{2H} + \left(\frac{b-a}{2}\right)^{2H} - 2(2^{2H-1} - 1)\left(\frac{b-a}{2}\right)^{2H}\right] = (1 - 2^{2H-2})\left(\frac{b-a}{2}\right)^{2H}
 \end{aligned}$$

Por tanto, para la k -ésima iteración, la varianza del desplazamiento del punto medio es igual a:

$$2^{-2kH} (1 - 2^{2H-2}) T^{2H}$$

La traza FBM generada con el algoritmo RMD puede ser interpretada como el proceso acumulativo de llegadas $A(t)$.

$$A(t) = M(t) + \sqrt{aM}Z(t) \quad (6.5)$$

donde M es la tasa media, y a es el factor de variabilidad definido como la relación de la varianza y el número medio de paquetes por unidad de tiempo. El proceso de incrementos (a veces denotado como proceso de tasa), da el número de llegadas por unidad de tiempo desde t a $t+1$.

$$A[t, t + \delta t] = M\delta t + \sqrt{aM}[Z(t + \delta) - Z(t)]$$

La entrada a la generación de $A(t)$ incluye M , a , y H , resultando en un modelo de tráfico parsimonioso de tres parámetros. Las ventajas de utilizar el algoritmo RMD para la generación de tráfico auto-similar son: simplicidad, eficiencia y rapidez.

Sus desventajas son:

1. Sintetizar a priori del tráfico deseado.
2. Imprecisión del parámetro H resultante del tráfico sintético.
Normalmente la estimación del parámetro H es superior a su valor real (ver capítulo 3) cuando éste pertenece al intervalo $[0,5 - 0,75]$, e inferior cuando el valor deseado es mayor que 0.75.
3. El valor de $A[t, t + \delta t]$ tiene que ser no-negativo; sin embargo, los resultados de la simulación (que representan el número de llegadas por unidad de tiempo) pueden ser negativos, siendo necesario truncar la salida. Ésto contribuye a la imprecisión en el parámetro H resultante.

El programa implementa este algoritmo para simular la realización del proceso estocástico FBM (*FBM (fractional brownian motion)*). Aunque este programa está escrito para un proceso FBM se puede convertir para simular cualquier proceso gaussiano cambiando las matrices de covarianza.

6.3. Modelo ON/OFF

El modelo de fuentes *on/off*, también llamado *packet train source model*, sirve para capturar las rafagas en el flujo de paquetes. Este método es particularmente atractivo porque evidencia el comportamiento de las fuentes de tráfico Ethernet.

Se supone un conjunto de fuentes que alternan periodos *on/off* y que exhiben el efecto "Noah" (síndrome de varianza infinita). El periodo *on* es aquel en el se transmite información a velocidad constante y el periodo *off* corresponde al tiempo entre transmisiones.

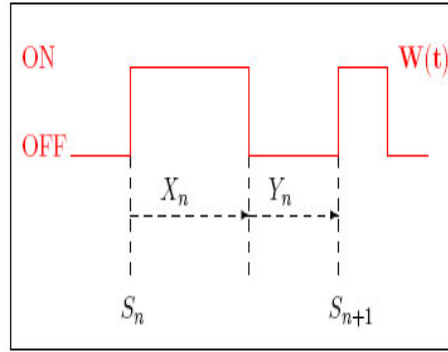


Figura 6.3: *Renewal reward process*

El efecto "Noah" garantiza que cada fuente exhibe características que cubren un amplio rango de escalas temporales y se puede describir utilizando distribuciones con varianza infinita. Matemáticamente se utilizan distribuciones heavy-tailed (de cola pesada) para incluir el efecto "Noah".

Para todas las fuentes los periodos *on/off* son independientes e idénticamente distribuidos. Asimismo las secuencias de estos periodos son independientes unas de las otras, por tanto no existe una alternancia estricta entre ellos. La distribución estadística de estos periodos puede tener varianza finita o infinita.

El modelado tradicional de tráfico asume que la varianza de las distribuciones de los periodos *on/off* es finita, por lo que la superposición de muchas fuentes equivale a ruido blanco; el flujo de tráfico agregado no presenta correlación a largo plazo.

6.3.1. Fuentes homogéneas

En primer lugar se va a suponer la existencia de una sola fuente y la atención se va a centrar en la serie de tiempo estacionaria $W(t), t \geq 0$ asociada.

$W(t)=1$: en el instante t se transmite información.

$W(t)=0$: en el instante t no se envía ningún paquete.

La serie $W(t)$ toma el valor 1 durante el periodo *on* y el valor 0 durante el periodo *off*. Cuando sólo existe una fuente los periodos *on/off* son estrictamente alternantes.

Los periodos *on* puede seguir diferentes distribuciones estadísticas pero siempre van a ser independientes.

Ahora se va a suponer la existencia de M fuentes i.i.d (independientes e idénticamente distribuidas). Cada una de ellas envía su propia secuencia de paquetes, por tanto se asocia a una serie $W^m(t), t \geq 0$ diferente. La superposición o cuenta acumulada de paquetes en el instante t es:

$$\sum_{m=1}^M W^{(m)}(t)$$

Rescalando el tiempo por un factor T , la cuenta de paquetes acumulados en el intervalo $[0, T]$ es:

$$W_M^*(Tt) = \int_0^{Tt} \left(\sum_{m=1}^M W^{(m)}(t) \right) du$$

Es de gran interés el comportamiento estadístico del proceso estocástico $W_M^*(Tt)$, $t \geq 0$ para un número muy elevado de fuentes y un periodo de tiempo grande. Este comportamiento depende de las distribuciones de los periodos de ON y OFF.

A partir del modelo FBM para el tráfico de paquetes agregado acumulado, (equivalentemente, modelo FGN para el tráfico agregado) se quiere elegir una distribución de tal forma que, para $M \rightarrow \infty$ y $T \rightarrow \infty$, $W_M^*(Tt)$ adecuadamente normalizado sea $\sigma_{lim} B_H(t), t \geq 0$, donde σ_{lim} es una constante positiva finita y $B_H(t)$ es el proceso estocástico FBM.

FBM es un proceso estocástico con media nula, incrementos estacionarios. Su función de covarianza es:

$$E[B_H(s)B_H(t)] = \frac{1}{2}(s^{2H} + t^{2H} - |s - t|^{2H})$$

Los procesos *Fractional gaussian noise (FGN)* corresponden a los incrementos $G_j = B_H(j) - B_H(j - 1) \quad \forall j \geq 0$. Su función de covarianza es:

$$E[G_H(j)G_H(j + k)] \sim H(2H - 1)k^{2H-2} \quad k \rightarrow \infty$$

donde $a_k \sim b_k$ significa $\frac{a_k}{b_k} \rightarrow 1$ a medida que k se aproxima a ∞ .

El decaimiento de la covarianza según una ley de potencia caracteriza la dependencia a largo plazo, *long range dependence*. Cuanto más alto es el valor de H más lento es el proceso de decaimiento.

Para especificar la distribución estadística de los periodos *on/off*

$$f_1(x), \quad F_1(x) = \int_0^x f_1(u)du, \quad F_{1c} = 1 - F_1(x)$$

$$\mu_1 = \int_0^\infty x f_1(x)dx, \quad \sigma^2 = \int_0^\infty (x - \mu_1)^2 f_1(x)dx$$

denota la función de densidad de probabilidad, la función de distribución acumulada, la distribución complementaria, la longitud media y la covarianza de un periodo *on* y $f_2, F_2, F_{2c}, \mu_2, \sigma_2^2$ son las correspondientes a un periodo *off*.

Se asume que cuando $x \rightarrow \infty$

$$F_{1c}(x) \sim l_1 x^{-\alpha_1} L_1(x) \quad 1 < \alpha_1 < 2 \quad o \quad \sigma_1^2 < \infty$$

y

$$F_{2c}(x) \sim l_2 x^{-\alpha_2} L_2(x) \quad 1 < \alpha_2 < 2 \quad o \quad \sigma_2^2 < \infty$$

$l_j > 0$ es una constante y $L_j > 0$ es una función tal que $\lim_{x \rightarrow \infty} \frac{L_j(tx)}{L_j(x)} = 1 \quad \forall t > 0$.

La media μ_j es siempre finita pero la varianza σ_j^2 es infinita cuando $\alpha_j < 2$. Por ejemplo, F_j podría ser Pareto, $F_{jc}(x) = k^{\alpha_j} x^{-\alpha_j} \quad \forall x \geq k \geq 0, \quad 1 < \alpha_j < 2$ e igual a 0 para $x < k$, o podría ser exponencial.

Para establecer el resultado principal es necesario introducir la notación utilizada.

Cuando $1 < \alpha_j < 2$, $a_j = l_j(\Gamma(2 - \alpha_j))/(\alpha_j - 1)$.

Cuando $\sigma_j^2 < \infty$, $\alpha_j = 2 \quad L_j = 1 \quad y \quad a_j = \sigma_j^2$. Los factores de normalización y las constantes de límite dependen de si

$$\Lambda = \lim_{t \rightarrow \infty} t^{\alpha_2 - \alpha_1} \frac{L_1(t)}{L_2(t)}$$

es finita, infinita o nula.

Si $0 < \Lambda < \infty$, $\alpha_{min} = \alpha_1 = \alpha_2$,

$$\sigma_{lim}^2 = \frac{2(\mu_2^2 a_1 \Lambda + \mu_1^2 a_2)}{(\mu_1 + \mu_2)^3 \Gamma(4 - \alpha_{min})}, \quad L = L_2$$

Si $\Lambda = 0$ o $\Lambda = \infty$

$$\sigma_{lim}^2 = \frac{2\mu_{max}^2 a_{min}}{(\mu_1 + \mu_2)^3 \Gamma(4 - \alpha_{min})}, \quad L = L_{min}$$

min es el índice 1 si $\Lambda = \infty$ (si $\alpha_1 < \alpha_2$) y es el índice 2 si $\Lambda = 0$, *max* denota el otra índice.

Teorema 1 Para valores grandes de M y T , el proceso agregado y acumulado de paquetes $W_M(tT)^*$ se comporta estadísticamente como:

$$TM \frac{\mu_1}{\mu_1 + \mu_2} t + T^H \sqrt{L(T)M} \sigma_{lim} B_H(t) \quad (6.6)$$

donde $H = (3 - \alpha_{min})/2$.

Más concretamente,

$$\zeta \lim_{T \rightarrow \infty} \zeta \lim_{M \rightarrow \infty} T^{-H} L^{-1/2}(T) M^{-1/2} \langle W_M^*(tT) - TM \frac{\mu_1}{\mu_1 + \mu_2} t \rangle = \sigma_{lim} B_H(t)$$

donde $\zeta \lim$ significa convergencia en el sentido de distribuciones de dimensión finita.

Heurísticamente, el teorema 6.6 establece que el factor $TM(\mu_1/(\mu_1 + \mu_2))t$ provee la principal contribución para grandes valores de M y T . Las fluctuaciones alrededor de este nivel son dadas por $\sigma_{lim}B_H(t)$ escalado por el factor $T^H L(T)^{1/2} M^{1/2}$.

Es esencial que los límites sean desarrollados en el orden indicado.

$1 < \alpha_{min} < 2$ implica $\frac{1}{2} < H < 1$, dependencia a largo plazo. Por tanto, el principal ingrediente para obtener un parámetro H mayor de 0.5 es la propiedad heavy-tailed de los periodos *on/off*; ésto es, una cola hiperbólica (decaimiento según una ley de potencia) para las distribuciones estadísticas de estos periodos con α entre 1 y 2.

$$F_{jc} \sim l_j x^{-\alpha_j} L_j(x) \quad x \rightarrow \infty, 1 \leq \alpha_j \leq 2$$

6.3.2. Comprobando el efecto "Noah"

A continuación se van a presentar las técnicas utilizadas para determinar la presencia o ausencia del efecto "Noah" para los periodos *on/off* derivados del tráfico de datos generado por las fuentes individuales o los pares fuente-destino. En el caso donde los datos son consistentes con el efecto "Noah", estas técnicas constituyen procedimiento rápidos para estimar la intensidad α de tal efecto. Para determinar la presencia o ausencia del efecto "Noah" en un conjunto de datos, se aplica logaritmo a ambos lados de la relación anterior, obteniendo:

$$\log F_{jc}(x) \sim \log(l_j) - \alpha_j \log(x) \quad x \rightarrow \infty \quad 1 < \alpha_j < 2 \quad (6.7)$$

La función de distribución complementaria de una muestra que presumiblemente se ha obtenido de una distribución que exhibe colas hiperbólicas, resulta en una línea aproximadamente recta para valores altos de x , con una pendiente de $-\alpha$, $1 < \alpha < 2$. Para ilustrar la efectividad de esta técnica aplicada a los datos de tráfico de la subred del departamento de Ingeniería Telemática, se va a centrar la atención en un par fuente-destino (otros producen resultados similares).

Un método estadísticamente más riguroso para estimar la intensidad del efecto "Noah" es conocido como *estimador de Hill*.

U_1, U_2, \dots, U_n denotan los periodos *on* y $U_{(1)} < U_{(2)} < \dots < U_{(n)}$ corresponde a sus estadísticos.

$$\alpha_n = \left(\frac{1}{k} \sum_{i=0}^{k-1} \log U_{(n-i)} - \log U_{(n-k)} \right)^{-1} \quad (6.8)$$

donde la elección de $1 < k \leq n$ indica cuántas observaciones grandes entran en el cálculo de la fórmula.

La representación gráfica del estimador Hill es una función de k (para un amplio rango de valores). Si el comportamiento de los datos es consistente

con la ecuación 6.8, la representación varía considerablemente para pequeños valores de k , pero se estabiliza a medida que se van incluyendo los datos de la cola de la distribución.

Un comportamiento en línea recta para pequeños valores de x en la gráfica de la función complementaria corresponde a una región de k valores donde el estimador Hill queda estable. En ausencia de este comportamiento lineal, la estima Hil continuará decreciendo a medida que k aumenta, un fuerte indicio de que los datos no son consistentes con la asunción de cola hiperbólica.

Aunque se conocen muchas propiedades teóricas del estimador Hill, éstas requieren asunciones extra sobre la distribución subyacente que no se pueden verificar en la práctica. Es especialmente cierto para la propiedad de normalidad asintótica del estimador Hill, que se utiliza para computar intervalos de confianza.

Por este motivo se van a elegir heurísticos para especificar rangos en el estimador Hill, no intervalos de confianza que son exactos teóricamente pero suponen condiciones que no se pueden verificar para el conjunto de datos en estudio.

6.3.3. Propiedad de robustez del efecto "Noah"

Antes de comprobar la presencia del efecto "Noah" en las trazas de tráfico generadas por los pares fuente-destino, se va a destacar la característica de robustez del efecto "Noah" que debería ser suficiente para resaltar el uso de los modelos *on/off* o *packet train* en el nivel de fuente o fuente-destino. En el pasado, el rechazo hacia este modelo se debía a la falta de interpretación física para definir objetivamente la noción de *periodo off*. El problema es decidir de una manera coherente la distancia apropiada entre trenes, decidir cuando ocurre la "salida" del tren anterior y la "llegada" del siguiente. Aquí se demuestra que, por lo que se refiere al "efecto Noah", no importa cómo hayan sido definidos los periodos *off* o distancias entre trenes (y por tanto, los periodos *on* o longitud del tren de paquetes).

El efecto "Noah" es robusto bajo un amplio rango de criterios para el valor umbral de t , que define explícitamente el periodo *off* como cualquier intervalo de longitud t segundos o más durante el que no se produce la llegada de ningún paquete.

La razón detrás de esta intensidad del efecto "Noah" en los periodos *off* para diferentes valores de t , es la propiedad de escala de las distribuciones que satisfacen la condición de cola hiperbólica.

Por propiedad de escala se entiende que si la distribución de la variable aleatoria U satisface la ecuación 8.2 y t denota un valor umbral,

$$P(U > u | U > t) \sim \left(\frac{u}{t}\right)^{-\alpha} \quad u > t \quad 1 < \alpha < 2 \quad (6.9)$$

El comportamiento de cola de las distribuciones de U condicionadas a diferentes valores del umbral t , difiere sólo en un factor de escala, por tanto da

lugar a representaciones de distribución complementaria (escala logarítmica) con pendiente idéntica.

La propiedad de robustez del efecto "Noah" se aplica también para los periodos *on*, aunque para explicar la intensidad del efecto "Noah" se tiene que recurrir a argumentos diferentes. Por tanto, mientras no haya una división natural en los periodos *on/off* en el nivel de fuente, tal división aparece en un amplio rango de escalas temporales; además, estas divisiones aparecen de una manera consistente.

6.3.4. Naturaleza del tráfico generado por un usuario Ethernet individual

Históricamente era necesario modelar de forma precisa las fuentes de tráfico para obtener un modelado exitoso del tráfico agregado, pero en el caso del tráfico de paquetes auto-similar, el conocimiento de las características fundamentales del tráfico agregado puede proveer nueva luz sobre la naturaleza del tráfico generado por un usuario individual.

En este apartado se da una explicación física de la propiedad de auto-similitud del tráfico LAN Ethernet en términos del comportamiento de los usuarios Ethernet individuales.

Se va a recurrir a la construcción de Mandelbrot de los procesos FBM (*fractional brownian motion*) empleando *renewal reward processes* como simples bloques de construcción e interpretándolo en el contexto de posibles comportamientos de los usuarios Ethernet.

El proceso *renewal reward process* $W^{(m)} = (W^{(m)} : t = 0, 1, 2, \dots)$ representa la cantidad de información (bits, bytes, paquetes) generada por el usuario m en el instante de tiempo t ($1 \leq m \leq M, t \geq 0$).

Dependiendo de si las unidades de información son bits (bytes) o paquetes, este modelo de fuente se convierte en la clase de modelos fluidos o en el modelo *packet train model*.

Para facilidad de presentación, se va a asumir que los "rewards" W_0, W_1, W_2, \dots toman valores 1 ó 0 (o para mantener $E[W] = 0$, +1 y -1 con la misma probabilidad), donde el valor 1/0 indica un periodo "activo"/"inactivo" en el que la fuente envía 1/0 unidades de información por unidad de tiempo. Aunque los sistemas de comunicación Ethernet no son *slotted* hay una unidad de tiempo natural (tamaño de slot) impuesto por la velocidad de transmisión de una unidad de información (60 μs para el paquete más pequeño (64 bytes)).

La propiedad crucial que diferencia el modelo de fuente *renewal reward process* del modelo fluido o del modelo *packet train* es que los intervalos *inter-renewal* (longitud de los periodos "activo"/"inactivo") se distribuyen según una variable aleatoria *heavy-tailed*, o utilizando la terminología de Mandelbrot, exhiben el **síndrome de varianza infinita**.

Aunque la propiedad *heavy-tailed* de los periodos "activo"/"inactivo" parece plausible a la luz de la manera en la que un usuario típico de una estación de trabajo contribuye al tráfico Ethernet, no se ha analizado el tráfico generado por los usuarios Ethernet individuales para validar este modelo de fuente.

En contraste, el modelo de fuente *renewal reward* para el tráfico generado por el usuario individual de una estación de trabajo es extremadamente simple

6.3.5. Implicaciones prácticas del efecto "Noah"

La propiedad de auto-similitud observada empíricamente permite una clara distinción entre los modelos tradicionales de tráfico y el tráfico coleccionado en la subred del departamento de Ingeniería Telemática.

La explicación física propuesta se basa en el efecto "Noah" y permite identificar la diferencia esencial entre los modelos de fuente *on/off* de ambos conjuntos; el modelado tradicional asume distribuciones de varianza finita para los periodos *on/off* (de hecho, se utilizan distribuciones exponenciales o geométricas), mientras que el modelado auto-similar se basa en la asunción del efecto "Noah", que requiere distribuciones de varianza infinita.

Desde un punto de vista más aplicado, cuestiones relacionadas con el impacto de la auto-similitud en la práctica (generación de tráfico real, rendimiento de la red, protocolos y controles) se puede reducir a la cuestión más básica de las implicaciones prácticas del efecto "Noah".

A continuación se va a presentar su impacto en el modelado de tráfico, generación de tráfico sintético y análisis de rendimiento de red.

6.3.6. Generación y modelado de tráfico

No hay duda de que el tráfico de la subred del departamento de Ingeniería Telemática es complejo. Se podría pensar que el modelo matemático debe ser igualmente complicado, debe tener muchos parámetros para capturar de forma realista la complejidad observada, pero no es así. Los modelos *on/off* modelan de forma parsimoniosa el tráfico de red, y proveen una explicación física del fenómeno de auto-similitud simple y consistente con las medidas de tráfico.

El deseo de una explicación fenomenológica de la auto-similitud del tráfico LAN ha resultado en nuevos descubrimientos acerca de la naturaleza del tráfico. Se identifica el efecto "Noah" como un ingrediente esencial, por tanto se va a describir una importante característica del tráfico por un parámetro único, intensidad α del efecto "Noah" en los periodos *on/off*.

Explicar y por tanto modelar el fenómeno auto-similar en el contexto de tráfico de datos en términos de la superposición de muchas fuentes *on/off* con distribuciones estadísticas de varianza infinita para las longitudes de los periodos *on/off* conduce a un método para generar largas trazas de tráfico

auto-similar en un tiempo razonable - asumiendo un entorno de computación paralela; se deja que varios procesadores generen tráfico de acuerdo al mismo modelo *on/off* (mismo valor de α); añadiendo (agregando) las salidas de todos los procesadores se obtiene tráfico auto-similar.

El objetivo es "verificar" experimentalmente el teorema 6.6 en el contexto de las medidas de tráfico realizadas en la subred del departamento de Ingeniería Telemática. La función de densidad de probabilidad de Pareto de parámetros α y β viene dada, como se vio anteriormente, por la expresión:

$$f(x) = \frac{\beta \alpha^\beta}{x^{1+\beta}} \quad (6.10)$$

Los parámetros de las fuentes *on/off* de Pareto se ajustarán del siguiente modo:

- La media de las distribuciones que describen los periodos *on/off* y la tasa de transmisión en el estado *on* λ , de la misma manera que el modelo clásico de tráfico.
 - El parámetro β se ajusta a los valores observados en el tráfico, que está en valores que oscilan entre 0.95 y 1.25. ($H = 0.6$)
1. Se parte de un generador de números aleatorios según una distribución uniforme U entre $[0,1]$.
 2. Se iguala la distribución U a la función de distribución de la variable aleatoria de Pareto, cuya expresión es:

$$F(x) = P(X \leq x) = 1 - \frac{\alpha^\beta}{x}, \quad x \geq \alpha$$

y se obtiene

$$\begin{aligned} U &= 1 - \frac{\alpha^\beta}{x} \rightarrow \\ \frac{\alpha^\beta}{x} &= (1 - U)^{\frac{1}{\beta}} \rightarrow \\ x &= \alpha(1 - U)^{\frac{-1}{\beta}} \rightarrow \\ x &= \alpha(U)^{\frac{-1}{\beta}} \end{aligned}$$

6.3.7. Análisis de rendimiento

Los beneficios prácticos de modelar el tráfico medido con pocos parámetros son especialmente importantes cuando la atención se centra en los impactos potenciales de las características del tráfico, efecto "Noah" y el efecto "Joseph", sobre el rendimiento de la red, el análisis de protocolos y el control de congestión.

Varios trabajos han demostrado que existe una diferencia significativa en el rendimiento (expresado en términos de la distribución de la longitud de cola) de un sistema modelado según procesos markovianos y aquellos que exhiben el efecto "Joseph". Más específicamente, mientras que la distribución de la longitud de cola decrece exponencialmente en el primer conjunto, en el segundo lo hace más lentamente (depende de la intensidad H del efecto "Joseph").

En la práctica, no capturar el efecto "Joseph" puede llevar a unas predicciones de rendimiento excesivamente optimistas y por tanto los requisitos de calidad de servicio (QoS) no se van a garantizar en un escenario real. Esta observación es de particular importancia en el contexto de ancho de banda equivalente (*equivalent bandwidth*). Al mismo tiempo, la presencia del efecto "Joseph" en el tráfico medido no excluye economías de escala (ganancias estadísticas por multiplexación) multiplexando un elevado número de fuentes.

A la vista de la explicación física de que el efecto "Joseph" en el tráfico LAN agregado es causado por el efecto "Noah" en las fuentes *on/off* individuales que generan el flujo agregado, comprender los impactos de este efecto sobre el rendimiento del sistema de colas es esencial.

6.4. Resumen y trabajo futuro

Tras la síntesis de tráfico "auténtico", cabe plantearse las siguientes cuestiones:

- ¿Con qué grado de confianza se puede asegurar que el proceso de interés es dependiente a largo plazo y no simplemente, no estacionario? Esta cuestión es crucial porque la no estacionariedad se exhibe de forma similar a la dependencia a largo plazo.

Un test básico para la estacionariedad es separar el conjunto de datos en dos mitades y estimar H de forma independiente. Las dos estimas deberían producir (con sus márgenes de error) valores comparables; de no ser así, significaría que H varía en el tiempo y por tanto, que el proceso subyacente es no estacionario.

En los entornos LAN se ha descartado el hecho de que los efectos no estacionarios sean la explicación de la dependencia a largo plazo.

- La dependencia a largo plazo, ¿es una propiedad de las fuentes de tráfico o sólo del tráfico agregado en un enlace de red?

Las diferentes fuentes de tráfico interactúan entre sí, compitiendo por el ancho de banda ¹. Estas interacciones prolongan los periodos *on*, periodos en los que se transmite tráfico, mientras que las tasas de

¹Esto es particularmente cierto para el tráfico TCP debido al mecanismo de ventana adaptativa.

transmisión se tienden a homogeneizar. Ambos efectos fortalecen la combinación entre el tráfico de red y el modelo *on/off* para generar tráfico auto-similar.

Hay que tener precaución cuando se asume que las fuentes de tráfico son bien modeladas utilizando procesos auto-similares.

En algunas situaciones una "fuente" de tráfico puede ser tráfico agregado de un enlace anterior, lo que llevaría a modelar ese tráfico como una fuente auto-similar. Pero, debido a que el tráfico es distorsionado por la dinámica de red, tal modelo sería incompleto incluso si el último tráfico medido en el enlace previo fuera auto-similar.

Además, incluso si una fuente de tráfico es auto-similar y el tráfico de nivel de enlace resultante también lo es, es posible que la relación entre ellos sea compleja: la dinámica de red puede alterar la media, la varianza, el parámetro Hurst y el carácter de dependencia a corto plazo en la fuente; Ésto es, se podría alterar el proceso entero.

Algunos autores han presentado evidencia de que las fuentes Ethernet se pueden modelar como procesos *on/off*, lo que produce auto-similitud asintótica cuando estas fuentes son agregadas.

En este trabajo no se provee una guía del modelado individual de las fuentes.

- Relacionado con el punto anterior, si el objetivo de sintetizar tráfico auto-similar es utilizarlo como tráfico de *background*, se debe reconocer que debido a la dinámica de red no es posible separar claramente el tráfico de *background* del tráfico introducido.

Si se produce algún cambio en los parámetros de tráfico, la incorporación en el tráfico de *traffic* queda como un problema abierto; ésto es, ¿pueden ser modificados los métodos FFT o RMD para extrapolar el tráfico alterado después de estos cambios?. Incluso el conocimiento de qué cambios necesitan ser incorporados (media modificada, varianza y parámetro de Hurst) es un problema abierto.

- Es crucial comprender la importancia relativa de un proceso de llegadas dependiente a corto plazo de uno dependiente a largo plazo. No hay un balance fijo entre los dos, para algunas situaciones puede dominar SRD o LRD y para otras cada uno de ellos podría contribuir con efectos diferentes importantes.

Por ejemplo, cuando se desarrolla una simulación de un sistema de colas utilizando un buffer de tamaño finito, la fuerza de la SRD en la llegada de paquetes podría jugar un rol significativo en la distribución del retardo, mientras que la fuerza de la LRD influiría en los patrones de pérdida de paquetes.

Modelado de tráfico web

El *World Wide Web* fue diseñado e inicialmente desarrollado como un método de distribución de documentos científicos. Desde su pública aparición en 1992, el *World Wide Web* ha sido adoptado por usuarios de entorno comercial, educativo y gubernamental como un método de organizar y distribuir información fácilmente, que es rico en contenido multimedia: texto, gráficos, animación, audio y vídeo.

Actualmente, el web genera más tráfico de datos en Internet que cualquier otra aplicación. Como resultado, las características del web tienen implicaciones para la ingeniería de red y para la evaluación de rendimiento de internet. Debido al uso tan amplio del web, la comprensión de sus características resulta fundamental. En este capítulo se van a describir un número de características empíricas del web, centrando la atención en las medidas de las distribuciones de probabilidad.

Para ubicar las medidas en un contexto, se va a empezar por presentar cómo está organizado el web. La infraestructura básica utilizada por el web consiste en un conjunto de hosts (cada uno de ellos funciona como un cliente o un servidor o ambos) y una red a la que se conectan (normalmente se denomina Internet).

La extraordinaria popularidad del web parece surgir de una combinación de su utilidad y su facilidad de uso. Es útil como un medio para publicar y desarrollar información en una amplia variedad de formatos: texto, gráficos, animación, audio, vídeo e incluso software.

Su facilidad de uso se debe al ocultamiento de los detalles para contactar con sitios remotos en Internet, transportar datos en la red, formatear y mostrar la información solicitada independientemente del tipo de computadora particular.

La información del web consiste en un conjunto de ficheros identificados por una URL única. Sólo es necesario conocer la URL de un fichero para transferirlo desde donde esté almacenado (cualquier host de Internet) y mostrarlo en la computadora local.

El web está organizado utilizando un modelo cliente-servidor. Cada fichero es almacenado en un host, especificado como parte de su URL. Es-

tos hosts se denominan *servidores*. Cuando un usuario solicita un fichero, éste se transfiere al host local del usuario, *cliente*. El software empleado por el cliente para descargar y mostrar el contenido de los ficheros se llama *navegador*. La figura 7.1 muestra una vista esquemática de este modelo.

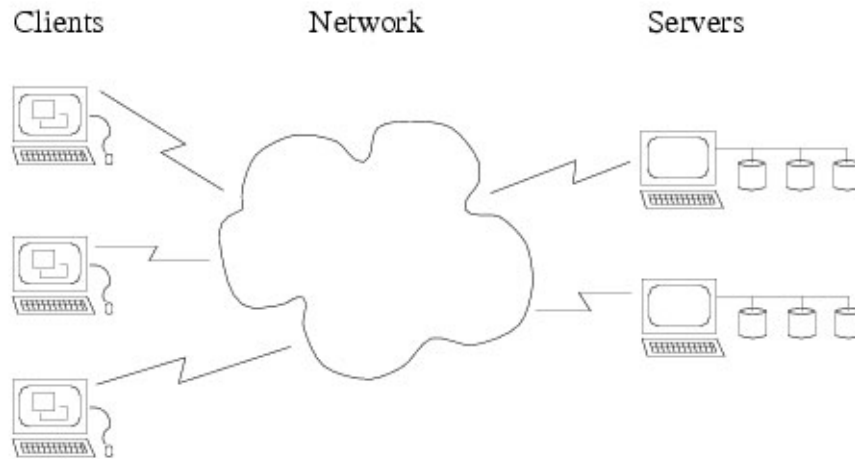


Figura 7.1: Clientes, red y servidores en el world wide web

Una vista más detallada, incluyendo detalles de implementación se muestra en la figura 7.2. La configuración mostrada es representativa de las conexiones entre un conjunto de hosts e Internet. Los clientes tienen conexión de una red de área local y ésta se conecta, a través de un gateway o router (computadoras especiales), se conecta a Internet.

Todos los clientes de una subred emplean *caching* para aumentar la velocidad de acceso a los ficheros disponibles en el web. Una caché es un conjunto de copias de ficheros ya accedidos, que se guardan en un dispositivo local de almacenamiento - memoria principal o disco. Cuando se realiza la petición de un fichero, el navegador del cliente comprueba si el archivo está en la caché; si lo está, no es necesario descargarlo a través de Internet sino que basta con copiarlo directamente de la caché. El tiempo de respuesta de ambos servicios varía en dos o tres órdenes de magnitud ¹

Para implementar algún mecanismo de caché, el navegador examina cada petición de URL. Si la petición puede servirse desde la caché, *cache hit*, si no *cache miss*.

Ya que el navegador dispone de una cantidad limitada de espacio para implementar la caché, se debe decidir el almacenamiento de un fichero para evi-

¹La mayoría de los navegadores implementan algún mecanismo de caché desde los primeros días del web

tar ocupar espacio innecesariamente. Este conjunto de decisiones se conoce como *gestión de políticas de caché*.

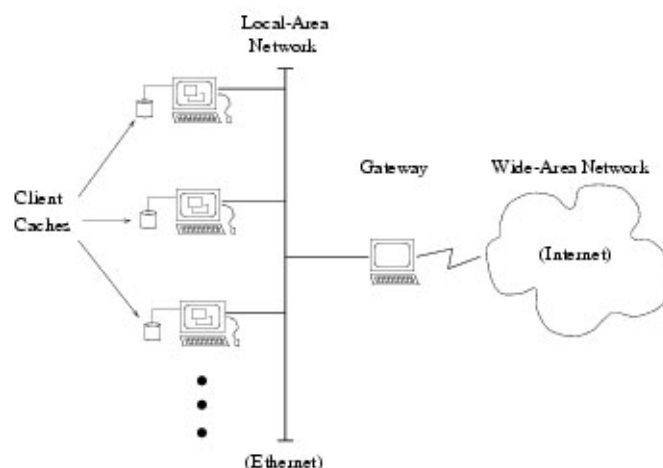


Figura 7.2: Implementación de las conexiones cliente al web

HTTP, el protocolo responsable de la transferencia del contenido web, domina las trazas de tráfico. Para caracterizar el comportamiento del tráfico HTTP se han propuesto algunos modelos analíticos; la mayoría de ellos describen el comportamiento de un cliente común con diferentes niveles de detalle. Un pequeño número de trabajos se centran en el comportamiento de una agregación de clientes web; la ventaja principal de estos modelos es la simplicidad.

En este trabajo se propone un nuevo modelo HTTP, que está basado en el concepto de *comportamiento agregado* y presenta dos ventajas principales: pequeño número de parámetros y control de carga preciso.

7.1. Background

En este apartado se presenta la información de HTTP que resulta de interés en el modelado de tráfico. De la misma forma se van a presentar las dos aproximaciones principales, cliente y agregación, y los beneficios e inconvenientes de cada uno de ellos.

7.1.1. Protocolo HTTP

HTTP es un protocolo petición-respuesta diseñado para la transferencia de ficheros web. Cada transferencia consiste en una petición de páginas u objetos por parte de un cliente a un servidor y la correspondiente respuesta o

notificación de error. De una forma más simple y general, el usuario solicita una URL (Universal resource locator) al navegador y desde el navegador se realiza la solicitud al servidor. Cuando se recibe la página, el navegador realiza un parseo sobre ella para encontrar y solicitar al servidor los objetos componentes (imágenes, sonidos, clases Java, ...). El usuario tiene ahora disponible la información y los enlaces a otras páginas; si el usuario selecciona cualquier link el proceso empieza de nuevo.

La primera versión HTTP fue HTTP/0.9 y consistía en un protocolo simple para la transferencia no refinada de datos a través de Internet. La siguiente versión mejoró el protocolo permitiendo que el formato de los mensajes fuera MIME, conteniendo en la semántica de la petición (o respuesta) meta-información sobre los datos transferidos y los modificadores. MIME permite a los navegadores mostrar información que no está en formato HTML como gráficos, audio y vídeo. Sin embargo, en HTTP/1.0 no se tenían en cuenta los efectos de la jerarquía de los proxies, el efecto de la caché, la necesidad de conexiones persistentes o hosts virtuales. Para cubrir estas deficiencias se diseñó la versión 1.1 del protocolo HTTP.

La comunicación HTTP se soporta normalmente sobre conexiones TCP/IP aunque puede ser implementado en otras arquitecturas de protocolos siempre que en el nivel de transporte se garantice una comunicación fiable. El puerto reservado para estas comunicaciones es el puerto 80 aunque se pueden utilizar otros. Ésto no impide que HTTP sea implementado sobre cualquier otro protocolo en Internet, o en otras arquitecturas de protocolos.

7.1.2. Modelado de tráfico HTTP

La simulación es una herramienta ampliamente utilizada para la evaluación de redes pero es importante disponer de modelos de tráfico para adquirir resultados útiles. La mayoría de los trabajos realizados sobre el modelado de tráfico web se ha concentrado en el desarrollo de *modelos cliente*, cuya atención se centra en el comportamiento de cada cliente por separado. Otra aproximación posible consiste en modelar el comportamiento agregado de varios clientes web, *modelo de agregación*.

Ambos modelos tienen ventajas e inconvenientes. El *modelo cliente* es capaz de capturar más detalles de la aplicación, es una mejor imitación. Sin embargo, este alto nivel de detalle supone una mayor complejidad en el modelo porque precisa la comprensión y configuración de más parámetros. En algunas situaciones el nivel de detalle no ayuda en la evaluación, por tanto muchos de ellos no importan. Algunos ejemplos de este tipo de evaluación son los desarrollados en los enlaces congestionados: provisión de recursos, comportamiento de cola, ... En estos casos los modelos más simples constituyen una mejor solución.

El *modelo de agregación* no es una aproximación tan cercana del tráfico real. A pesar de ésto, su simplicidad permite simular algunas condiciones e

identificar comportamientos que sería difícil con *modelos cliente*. Además, los *modelos cliente* tienden a consumir más recursos de computación que los *modelos de agregación* cuando se representa un número alto de clientes web en un entorno de simulación. En ambos modelos un asunto importante es la elección de las características requeridas por la aplicación, que son el centro del desarrollo del modelo. Algunos ejemplos de estas características son variabilidad, carga de red, dependencia a largo plazo, ...

Un modelo (*de agregación o cliente*) utiliza parámetros para reproducir propiedades concretas de la aplicación web. Algunos ejemplos de parámetros son: tamaño de transferencia, intervalos entre páginas, número de objetos por páginas, ... Para describir estos parámetros se utilizan dos aproximaciones: basada en muestras de tráfico real o analítica.

Los modelos creados utilizando estas aproximaciones son conocidos como modelos estructurales porque ellos intentan caracterizar la naturaleza del tráfico.

El uso de muestras de tráfico real consiste en describir un parámetro de aplicación concreto a través de un conjunto de valores predefinidos que se colección en un entorno de red real. La ventaja principal de este método es la facilidad de implementación y la representación precisa de un sistema conocido. sin embargo, esta aproximación trata el tráfico generado como una "caja negra". Nuevas condiciones o demandas variables no son fáciles de configurar en un generador de tráfico basado en este modelo.

La aproximación analítica se basa en el uso de distribuciones de probabilidad para describir un parámetro concreto ². Cuando la distribución es conocida, se permite generar nuevas y diferentes secuencias de valores siguiendo tal distribución. El inconveniente principal de esta aproximación es la dificultad encontrada en la identificación y configuración de la distribución que describe adecuadamente esta secuencia de valores aleatorios.

Una tercera aproximación puede ser incluida, que consiste en utilizar procesos abstractos conocidos para intentar capturar únicamente las propiedades estadísticas del tráfico con independencia de los mecanismos subyacentes de la generación del tráfico. Esta aproximación es eficiente y su implementación es bastante simple. Además, esta aproximación es útil cuando son de interés características específicas. Por ejemplo, la auto-similitud es fácilmente reproducida por un proceso FBM *fractional brownian motion*. Sin embargo, este tipo de métodos no captura factores de importancia del perfil de tráfico y descuida elementos tales como el control de congestión de TCP, que es una característica de importancia del tráfico HTTP.

²Una distribución de probabilidad describe el comportamiento de una secuencia de valores aleatorios dado un conjunto de muestras suficientemente alto

7.1.3. Colección del tráfico

El desarrollo de modelos de tráfico para aplicaciones interactivas supone un estudio del tráfico efectivo generado por estas aplicaciones. En otras palabras, para desarrollar un modelo de tráfico HTTP es importante coleccionar muestras del entorno real y extraer la información necesaria, aunque en ocasiones esta recuperación no es trivial y se requiere el desarrollo de sofisticados heurísticos. Los principales métodos para coleccionar tráfico HTTP son:

Logs de servidor: son generalmente fáciles de obtener porque uno de los roles del servidor es asistir el impacto en el sistema. Como resultado la mayoría de los servidores mantienen registros detallados de cada uno de los accesos realizados.

Por otro lado, existe una gran dificultad para capturar patrones de acceso en varios servidores web y la información de las cabeceras HTTP no está disponible.

Logs de cliente: el cliente típicamente no registra sus actividades. Desafortunadamente, es difícil utilizar registros de servidores para obtener una imagen de la actividad web en una red de área local, ya que cada cliente visita diferentes servidores en un pequeño espacio de tiempo. Los logs del cliente presentan una manera difícil de identificar el comportamiento agregado.

Logs de proxies: se basa en la colección de información de una manera similar a la captura de paquetes. Una de las desventajas es su limitación, un proxy normalmente sirve a una LAN o un pequeño grupo de ellas, mientras que la captura de paquetes se puede ejecutar en una red de tránsito entre varias otras red.

Trazas de paquetes: se coleccionan aquellos paquetes que transportan tráfico HTTP en la subred del Departamento de Ingeniería Telemática. La única desventaja es la dificultad en la interpretación de algunos parámetros tales como el número de objetos efectivos transferidos por página. A pesar de ello, en este trabajo se va a utilizar este método.

Después de capturar el tráfico es necesario identificar y capturar los parámetros de interés, que varían en cada modelo. Algunos son medidos fácilmente y otros requieren técnicas sofisticadas.

7.2. Modelo de tráfico

Un buen punto de partida para el desarrollo del modelo es establecer su objetivo, qué aplicación describirá y cuáles son las características que se pretenden modelar. Así que es importante establecer un perfil antes de empezar el desarrollo del modelo; a pesar de su simplicidad, esta metodología

puede minimizar el tiempo de desarrollo.

El modelo debe tener poco parámetros. Además, el generador de tráfico basado en este modelo debería ser utilizado como entrada a enlaces congestionados. Así que, el modelo podría ignorar algunos detalles relacionados con los clientes web individuales porque el comportamiento agregado se mantiene.

Para los generadores de tráfico (también llamados generadores de carga), generalmente no existe una manera simple de ajustar la carga. Es común utilizar la carga media generada por un conjunto de clientes para especificar la configuración de red.

De acuerdo con la teoría de colas, el concepto de carga o factor de utilización puede ser escrito como:

$$\rho = R/C$$

donde

R: tasa de llegada de conexiones TCP y

C: tasa máxima o capacidad del sistema.

Bajo la suposición de que en la subred del Departamento de Ingeniería Telemática existe un único servidor, la expresión anterior se convierte en:

$$\rho = \lambda \bar{x}$$

donde

λ es la tasa media de llegada de conexiones TCP y

\bar{x} es el tiempo medio de servicio (duración media de las conexiones TCP).

Considerando el contexto de tráfico HTTP,

$$\bar{x} = \bar{L}/C$$

donde

\bar{L} es el tamaño medio de transferencia y

C es la tasa máximo o capacidad del enlace.

ρ es un parámetro de ajuste y se utiliza para elegir diferentes condiciones de carga, C es un valor fijo para una configuración de red concreta y \bar{L} controla el tamaño medio de una transferencia web. Por último, λ describe la tasa de llegada de conexiones, que varía de acuerdo con \bar{L} para mantener el valor de ρ . Por tanto la tasa de llegadas se puede escribir como:

$$\lambda = \rho C / \bar{L}$$

Ya que $T = 1/\lambda$ describe el tiempo entre la llegada de dos conexiones consecutivas,

$$T = \bar{L} / \rho C$$

En las siguientes figuras se observa que la función de distribución de probabilidad acumulada de los tiempos de transmisión es *heavy-tailed*. Este comportamiento es especialmente importante porque apoya la explicación de la observada auto-similitud del tráfico de red y está relacionado con la distribución del tamaño de los ficheros transferidos.

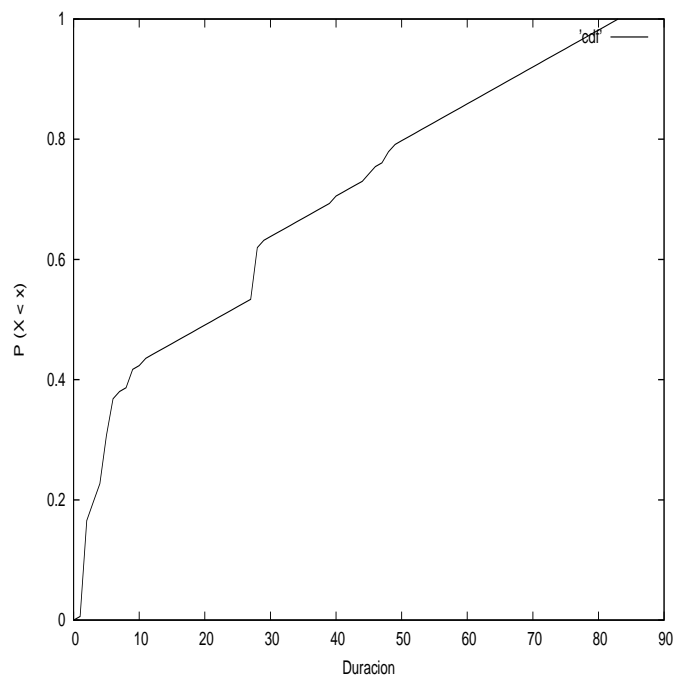


Figura 7.3: cdf (*cumulative distribution function*) de la duración de las conexiones www.

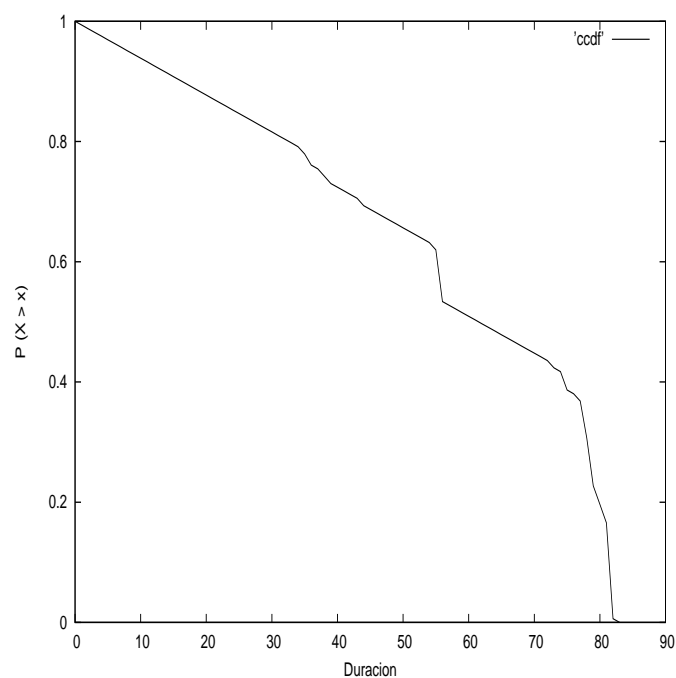


Figura 7.4: ccdf (*complementary cumulative distribution function*) de la duración de las conexiones www.

7.2.1. ¿Por que los tiempos de transmisión se distribuyen según una variable aleatoria *heavy-tailed*?

Para comprender por qué los tiempos de transmisión *heavy-tailed* se debe examinar la distribución estadística del tamaño de los ficheros transferidos.

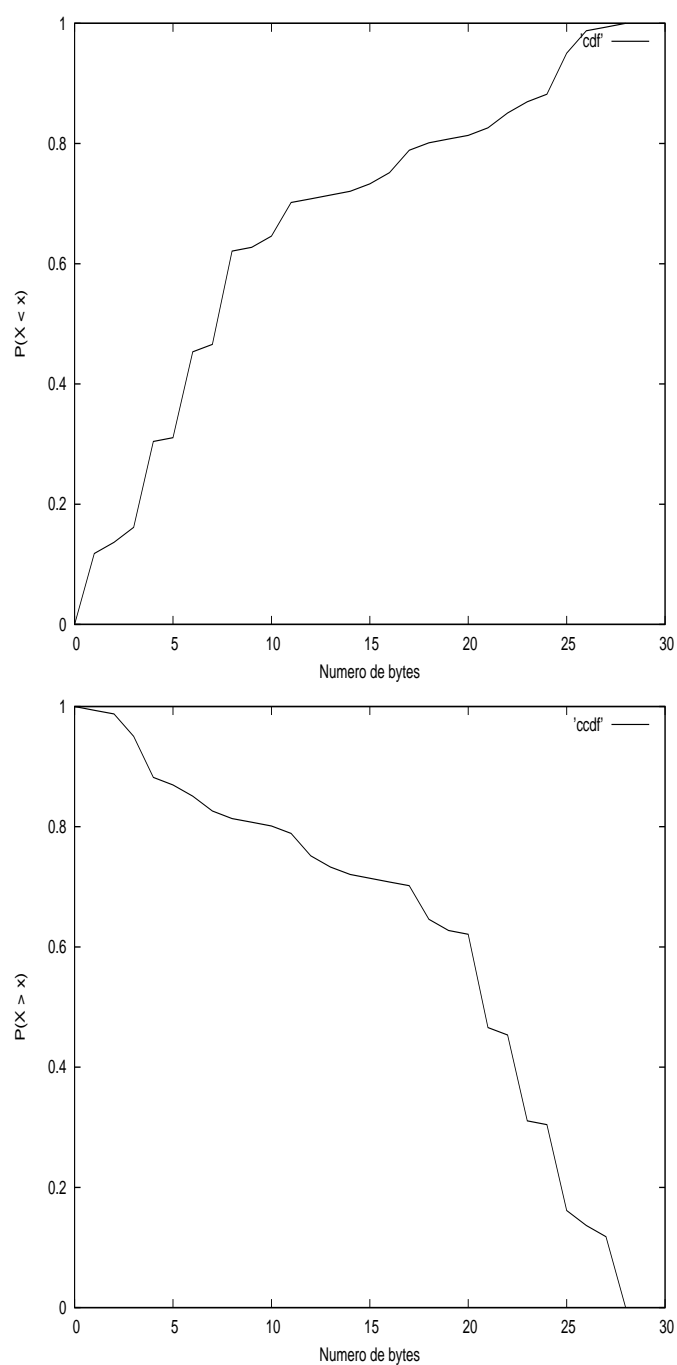


Figura 7.5: cdf (*cumulative distribution function*) y ccdf (*complementary cumulative distribution function*) del número de bytes transmitidos en las conexiones www.

La representación anterior corresponde a la función de distribución de probabilidad de los bytes transmitidos en las conexiones TCP. Se demuestra que la distribución estadística que mejor caracteriza esta variable es una distribución de cola pesada (distribución de Pareto).

Una cuestión importante es **por qué** la transferencia de ficheros sigue una distribución *heavy-tailed*. Por un lado, está claro que la petición de ficheros constituye una entrada del usuario al sistema y por tanto es natural asumir que ésta podría ser la responsable principal de la distribución estadística del tamaño de los ficheros transferidos. Si fuera realmente así, cambios en el comportamiento del usuario afectarían a la naturaleza *heavy-tailed* del tamaño de los ficheros, y por implicación, a las propiedades auto-similares del tráfico de red.

Se va a demostrar que la petición de los ficheros no es *intrínsecamente* responsable de la naturaleza *heavy-tailed* de las transferencias, sino que tal naturaleza está determinada por el tamaño de los ficheros disponibles en el web.

7.2.2. La naturaleza del conjunto de ficheros únicos y la acción de caché infinita

Tres de los conjuntos más importantes que se han coleccionado son:

- El conjunto de peticiones: es un registro de todas las peticiones de URL realizadas por los usuarios. Estos conjuntos contienen muchas peticiones duplicadas, que pueden ocurrir cuando un usuario solicita un fichero más de una vez, o cuando más de un usuario pide el mismo fichero. Muchas peticiones resultan en accesos hit a la caché, que significa que son satisfechas sin generar tráfico en la red
- El conjunto de transferencias de fichero: ésto consiste de todos los accesos "miss" de la caché. Cada elemento del conjunto corresponde a una instancia de un fichero que está siendo transferido en la red. Así que es un subconjunto adecuado de un conjunto de peticiones de fichero. Sin embargo, a pesar de la acción de caché, algunos ficheros son todavía transferidos por la red más de una vez, así que este conjunto también contiene ficheros duplicados.
- El conjunto de ficheros únicos: este conjunto contiene exactamente una entrada de cada fichero, independientemente cuántas veces haya sido solicitado o transferido. Así que este conjunto es también un subconjunto adecuado de los dos anteriores.

El argumento que se emplea en este trabajo se basa en dos importantes propiedades del conjunto de ficheros únicos:

1. El conjunto de ficheros únicos es el límite del conjunto de ficheros transferidos cuando el tamaño de la caché tiende a infinito.

Es de utilidad imaginar el comportamiento de una caché de infinito tamaño, que no rechazaría el almacenamiento de ningún fichero por falta de espacio en memoria o en disco; las únicas referencias que causarían *cache misses* serían aquellas que ocurren la primera vez.

2. El conjunto de ficheros disponibles es el límite del conjunto de ficheros únicos cuando el tiempo tiende a infinito.

Se considera de nuevo la acción de una caché infinita. Las peticiones de ficheros resultan *cache misses* si no han sido procesados anteriormente por la caché. Si se asume que los usuarios continúan realizando peticiones nuevas (ficheros que no se han descargado anteriormente), aumentará la fracción de ficheros disponibles accedidos por el usuario. A medida que el tiempo avanza, la fracción de ficheros visitados crece tanto que constituye un subconjunto apreciable de los ficheros disponibles. Cuando esto ocurre, es razonable esperar que el subconjunto de ficheros visitados se aproxima en distribución al conjunto de ficheros disponibles. Ya que cada uno de los ficheros visitados ocurre exactamente una vez en el conjunto de ficheros únicos, este conjunto tiende a aproximarse en distribución al conjunto de ficheros disponibles. Por supuesto, una cuestión importante relacionada con este argumento es si ha transcurrido tiempo suficiente desde el inicio de la conexión web para que ocurra este efecto.

Se debe apuntar que si los usuarios visitan nuevos ficheros de forma aleatoria, se necesita muy poco tiempo para que ocurra este efecto, por tanto el conjunto de ficheros únicos sería desde el comienzo una muestra aleatoria del conjunto de ficheros disponibles.³

Estas dos propiedades sugieren que para sistemas con cachés de gran capacidad que están activos durante periodos largos, el conjunto de ficheros transferidos se aproximará naturalmente al conjunto de ficheros disponibles - y será relativamente independiente de las solicitudes particulares de los usuarios. El objetivo de cualquier gestión de políticas de caché es intentar simular una caché infinita con recursos limitados. Cuanto mejor es el rendimiento de una caché real, más próximo estará el conjunto de ficheros transferidos al conjunto de ficheros únicos.

Los tiempos entre llegada de conexiones TCP cubren un amplio rango de valores en aplicaciones tales como el web. Los tiempos más pequeños se crean cuando una página web tiene múltiples imágenes embebidas porque la

³Por simplicidad se asume que el conjunto de ficheros disponibles es estático. Aunque se mantiene en continuo crecimiento, el argumento presentado aquí mantiene que: 1) los usuarios visitan nuevos ficheros a una velocidad mayor que la velocidad a la que crece el conjunto de ficheros disponibles o 2) el patrón del usuario que visita nuevos ficheros es independiente de sus tamaños

descarga de cada una de ellas se lleva a cabo en una conexión HTTP diferente (en la versión 1.0 del protocolo HTTP se realizaba la transferencia de todos los objetos web en la misma conexión, conexión persistente).

Una vez un usuario empieza a navegar por el web, visita un conjunto de páginas web; ésto genera tiempos entre llegada de conexiones que son de tamaño intermedio, pero pueden transcurrir periodos de tiempo en los que no haya actividad y por tanto se creen tiempos entre llegadas grandes.

El conjunto de ficheros transferidos es un superconjunto de los ficheros único, pero ambos grupos tienden a igualarse a medida que aumenta la capacidad de la caché. Ésto significa que el conjunto de ficheros transferido se espera en general para ser similar a la distribución de los ficheros disponibles en el web y, por el efecto de la caché, relativamente insensible a las peticiones particulares realizadas por los usuarios.

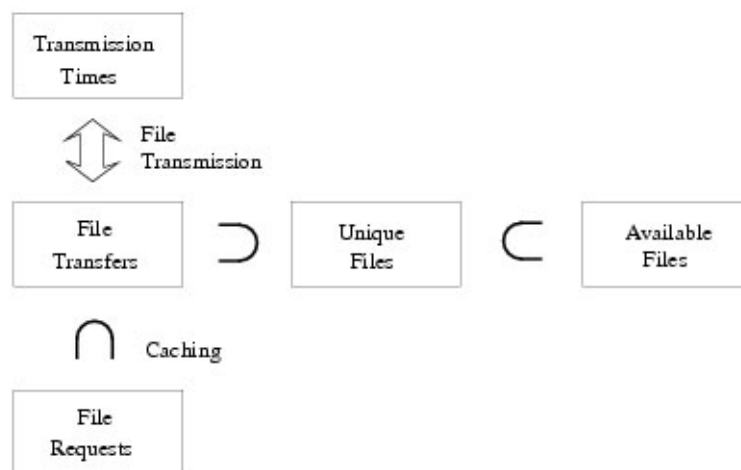


Figura 7.6: Relación entre los conjuntos de datos

7.2.3. ¿Por qué los ficheros disponibles en el web son *heavy-tailed*?

Una posible explicación podría ser que los formatos multimedia fomentan un aumento en el tamaño de los ficheros y por ello aumenta el peso de la cola de la distribución. Observando la figura 7.6 no parece que ésta sea la causa principal del comportamiento *heavy-tailed* de la distribución estadística del tamaño de los ficheros transferidos en el web.

La figura 7.7 se ha construido a partir del conjunto de ficheros disponibles en los servidores web, clasificando los ficheros en siete categorías diferentes: imágenes, audio, vídeo, texto, archivos, texto preformateado y ficheros com-

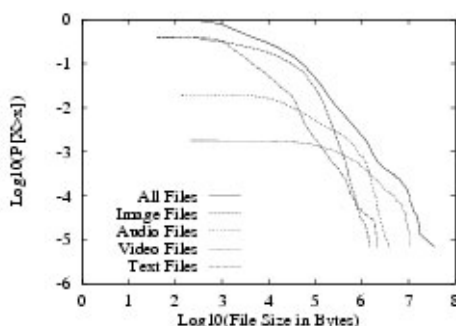


Figura 7.7: CD del tamaño de los ficheros de 32 sitios web

primidos. Esta simple clasificación permite recoger el 85 por ciento de todos los ficheros disponibles. Las categorías de imágenes, audio, vídeo y texto capturan el 97 por ciento del subconjunto anterior. La distribución acumulada de estas cuatro categorías, expresada como una fracción del conjunto total de ficheros, se muestra en la figura 7.7.

La línea superior de la figura es la distribución de todos los ficheros. Las tres líneas intermedias, son los componentes de esa distribución atribuible a imágenes, audio y vídeo respectivamente. La línea de la parte inferior es la componente atribuible a texto (HTML).

La figura 7.6 muestra que el efecto de añadir ficheros multimedia al conjunto de ficheros de texto es incrementar el peso de la cola. Sin embargo, ésto sugiere también que la distribución de los ficheros de texto podría ser en sí misma *heavy-tailed*. Utilizando la técnica de ajuste de mínimos cuadrados para las porciones de la distribución en las que $\log_{10}(x) > 3$, se llega a que para todos los ficheros disponibles $\hat{\alpha} = 1,06$ pero que para los ficheros de texto solo $\hat{\alpha} = 1,36$. Los efectos de los varios tipos multimedia también son evidentes a partir de la figura.

En el rango de 1.000 - 30.000 bytes, el peso de la cola aumenta por la transferencia de archivos gráficos.

En el rango 30.000 - 300.000 bytes, el peso de la cola aumenta por la transferencia de archivos de audio.

En el rango de más de 300000 bytes, el peso de la cola aumenta por la transferencia de archivos de vídeo.

Otro ejemplo que sugiere que el tráfico multimedia no es el responsable fundamental de la naturaleza *heavy-tailed* de los ficheros disponibles en el web es la comparación gráfica de la distribución de los ficheros disponibles en el web con la distribución completa de los ficheros de un sistema Unix.

La cola de la distribución de los ficheros disponibles en el web no es tan pesada como la cola de la distribución de los ficheros Unix.

Se comprueba que la tasa de transferencia (en ambos sentidos) sigue una distribución Weibull.

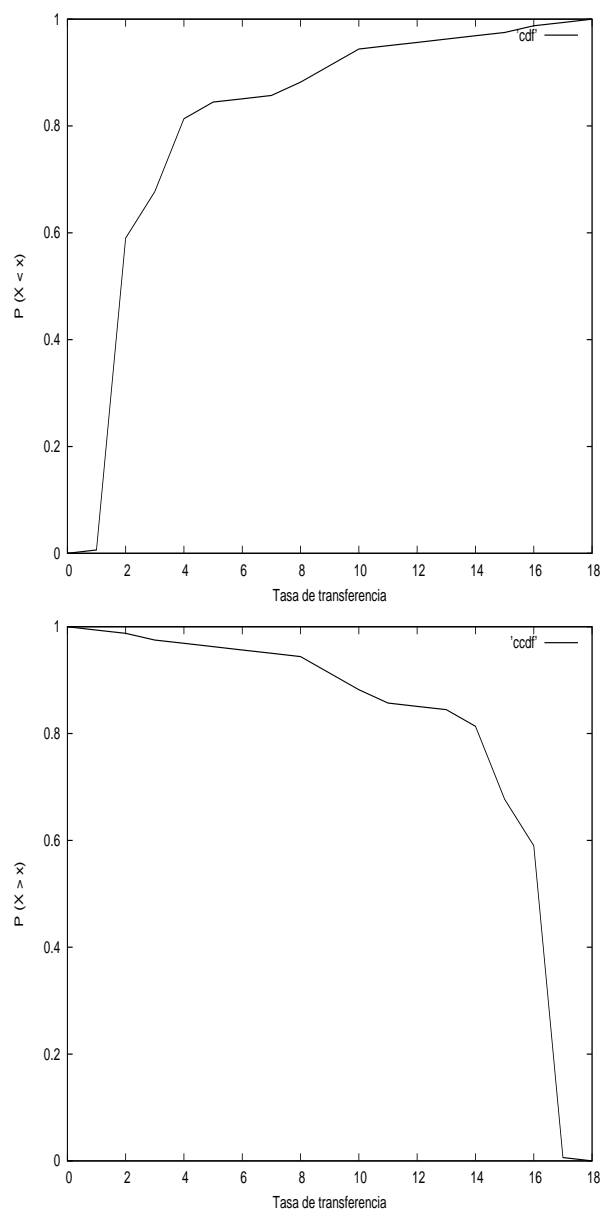


Figura 7.8: cdf (*cumulative distribution function*) y ccdf (*complementary cumulative distribution function*) de la tasa de transferencia de las conexiones www (*upstream*).

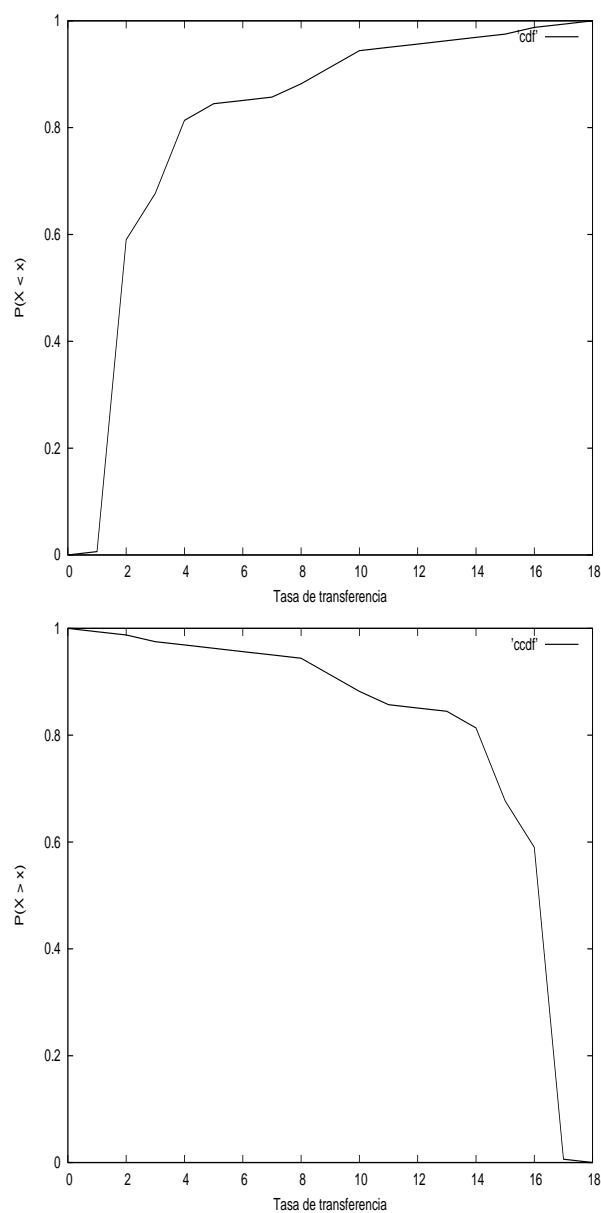


Figura 7.9: cdf (*cumulative distribution function*) y ccdf (*complementary cumulative distribution function*) de la tasa de transferencia de las conexiones www (*downstream*).

7.2.4. Implicaciones para el tráfico

La distribución *heavy-tailed* del tamaño de los ficheros disponibles en el web es la causa de la auto-similitud del tráfico www.

Algunos autores han demostrado que si el tráfico se construye como la suma de muchos procesos *on/off*, en los que los periodos *on/off* son independientes y se distribuyen según una variable aleatoria *heavy-tailed* con parámetro α , resulta una serie asintóticamente auto-similar con parámetro $H = (3 - \alpha)/2$. En el contexto del servicio *world wide web*, los procesos *on/off* individuales son análogos a las sesiones de los navegadores. En cada sesión se distinguen periodos de silencio o de recepción de los datos transmitidos a una tasa regular. Ésta es una simplificación de un entorno web real, pero indica que si la duración de las transmisiones es *heavy-tailed*, es muy probable que la naturaleza del tráfico resultante sea auto-similar. El grado de auto-similitud, medido con el parámetro H , está relacionado linealmente con el parámetro α de la distribución del tamaño de los ficheros.

En la siguiente tabla se muestra la información necesaria para caracterizar adecuadamente los parámetros del tráfico web.

| <i>Variable</i> | $E[X]$ | $Var[X]$ | $Min[X]$ | $Max[X]$ |
|--------------------------------|---------|-------------|------------|----------|
| Duración conexiones [segundos] | 7.35698 | 64.1251 | 0.00412893 | 143.536 |
| Bytes upstream | 744.224 | 553869 | 47 | 3253 |
| Bytes downstream | 1742.37 | 3.03586e+06 | 154 | 12117 |
| Tasa transferencia upstream | 2753.71 | 7.58289e+06 | 13.056 | 42380.8 |
| Tasa transferencia downstream | 8112.73 | 6.58164e+07 | 14.2275 | 71447.1 |
| Tiempo entre conexiones | 278.17 | 77378.7 | 0.617452 | 2834.47 |

Tabla 7.1: Estadísticos de las conexiones www.

Modelado de tráfico DHCP

El protocolo DHCP (*dynamic host configuration protocol*) automatiza el proceso de configuración de dispositivos nuevos y existentes en las redes TCP/IP. DHCP desarrolla muchas de las funciones que un administrador de red lleva a cabo cuando se conecta un host en una red. Reemplazar el manual de configuración por un programa aporta flexibilidad, movilidad y control a la configuración de los equipos de la red.

En este capítulo se demuestra que el tiempo que transcurre entre la llegada de dos paquetes de este protocolo se distribuye según una variable aleatoria *heavy-tailed*, por tanto el tráfico de este tipo exhibe características de auto-similitud.

8.1. Un primer intento de automatizar la configuración de dispositivos

En 1985, Bill Croft y John Gilmore diseñaron un protocolo llamado *Bootstrap Protocol* o *BOOTP*. La idea que había detrás de BOOTP era automatizar la configuración de los dispositivos de red, eliminando la necesidad de la configuración manual.

BOOTP requiere la creación de una tabla que contenga una lista de los clientes BOOTP, sus direcciones IP y otros parámetros de configuración necesarios. Un cliente envía una petición a la dirección broadcast de la red (esta petición será respondida por el servidor) para obtener su configuración.

Este protocolo se encarga únicamente de la configuración de red del dispositivo a partir de la definición estática para ese cliente en la base de datos del servidor. Las tareas restantes deben ser desarrolladas por el administrador de red.

Con la proliferación de DHCP, los administradores de red pueden elegir también el nivel de control que ellos quieren ejercer en la provisión de direcciones y pueden decidir si los clientes deben ser registrados antes de que se les asignen direcciones IP.

Una gran ventaja de DHCP es que permite conectar dispositivos móviles en diferentes puntos de la red en cualquier instante de tiempo. Si un equipo portátil tiene configurado un cliente DHCP se puede conectar en la red y utilizar inmediatamente los servicios ofrecidos sin requerir la intervención de un administrador de sistema.

Aunque el número de clientes sea reducido, la ventaja de no tener que configurar manualmente cada dirección IP individual puede ser significativo. Configurar un servidor DHCP simple para una única subred no debe suponer mucho más tiempo que configurar la dirección IP de un nuevo equipo.

8.2. Asignando direcciones IP utilizando DHCP

Una diferencia importante entre la manera de asignar direcciones de un servidor DHCP y un administrador de red es que DHCP impone un límite en cuánto tiempo se puede utilizar una dirección IP. El resto del protocolo es análogo a lo que hace un administrador de red: un cliente DHCP (se refiere al dispositivo, no al usuario) solicita una dirección IP. El servidor, empleando su conocimiento sobre la red y una lista de direcciones IP e identificaciones de los clientes, provee una.

La diferencia principal entre el protocolo DHCP y la asignación manual de direcciones es el alquiler (*lease*).

El protocolo DHCP adjudica una dirección IP a un cliente con un periodo de posesión definido. Para evitar que el alquiler de una dirección IP expire (el cliente deja de tener acceso a los servicios de red), el cliente debe renovarlo cada cierto tiempo.

8.2.1. Regeneración de direcciones con DHCP

DCHP impide el uso de las direcciones IP después de expirar el periodo de alquiler y posee un mecanismo para que los clientes continuen renovando sus alquileres cuando se conectan a la red; con ello DHCP permite una regeneración fiable de las direcciones IP que no están siendo usadas. Si un dispositivo se desconecta de la red durante un periodo largo, debe contactar con el servidor DHCP para obtener su configuración de red cuando se conecta de nuevo. Si la dirección es regenerada, se le da una nueva para que no exista conflicto en la provisión de direcciones.

8.2.2. Describiendo servicios de red con DHCP

DCHP distribuye direcciones IP e información de configuración opcional:

- Direcciones de los routers por defecto.

- Direcciones de los servidores DNS.
- Máscara de subred y dirección de broadcast.

8.2.3. Añadiendo servicios con DHCP

Cuando es necesario añadir o mover un servicio de red, se emplean los procesos de renovación regular de alquiler de direcciones para propagar esta nueva información (DHCP no permite actualizaciones instantáneas).

El administrador simplemente actualiza la configuración del servidor DHCP como apropiada. Si la dirección IP de un servicio cambia, entonces la configuración debe ser actualizada para reflejar tal cambio.

Cuando los clientes DHCP renuevan sus alquileres, automáticamente reciben esta nueva información y empiezan a utilizarla.

8.3. Problemas del protocolo DHCP

8.3.1. Carga del servidor

El servicio de nombres de la red del departamento de Ingeniería Telemática se encuentra en una máquina rápida y alta capacidad de memoria. Sin embargo, el servidor DHCP puede estar instalado en una máquina de menores prestaciones debido a la baja demanda de este servicio.

8.4. Estados del cliente

La operación del cliente DHCP es modelada en *RFC2131* por una máquina de estados (*state machine*), y su comportamiento es descrito por un diagrama de transición de estados (*state transition diagram*). Cuando un cliente no tiene una dirección IP válida se dice que está en estado **init**. La figura 8.1 presenta los estados en los que se puede encontrar el cliente y las transiciones entre ellos. Durante el proceso de configuración inicial, el cliente pasa a estado **selecting**; cuando obtiene una dirección IP pasa a estado **bound**. Cuando el cliente rearranca pasa a estado **init-reboot** y después confirma que su dirección IP es válida, pasando al estado **bound**. Si algún servidor DHCP de la red envía un mensaje **dhcpnack** al cliente porque el cliente se ha movido a una nueva red, el cliente vuelve al estado **init**.

Alguna vez antes de que el alquiler de la dirección IP expire, el cliente entra en el estado **renewing** e intenta ampliar el periodo de posesión de la dirección IP enviando un mensaje *unicast* al servidor que le proporcionó tal dirección. Si el cliente no recibe respuesta a su petición de renovación, entra en estado **rebinding** y envía un mensaje *broadcast* para ampliar su alquiler

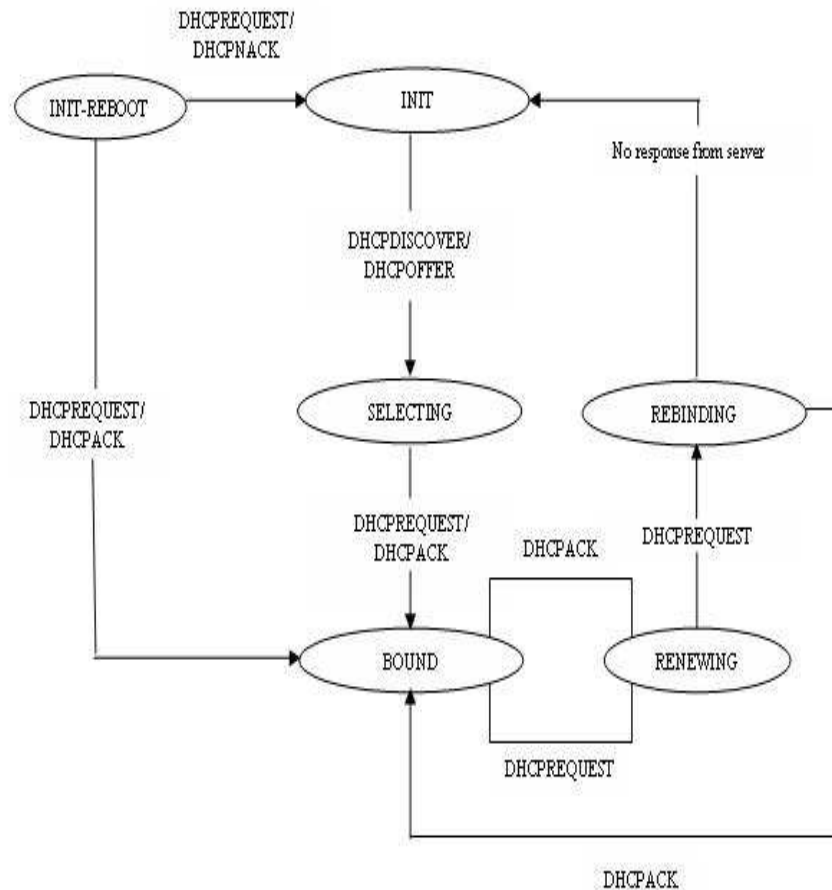


Figura 8.1: Una máquina de estados finitos para un cliente DHCP

de algún servidor disponible. Si el alquiler finalmente expira sin que el cliente haya podido renovarlo, vuelve al estado **init**.

8.4.1. Obteniendo una configuración inicial

La red a la que se conecta un equipo configurado para usar DHCP determina si tiene una dirección IP válida. Las razones por las que la dirección IP del cliente podría no ser válida son:

- El cliente se acaba de conectar a la red y no tiene una dirección IP asignada.
- El servidor DHCP ha informado al cliente que su dirección IP es inválida.

- El alquiler de su dirección previa ha expirado.

En todos estos casos, el cliente está en estado **init** porque no tiene una dirección IP válida.

Mensaje **DHCPDISCOVER**

Para obtener una dirección IP y otros parámetros de configuración el cliente debe localizar uno o varios servidores DHCP. A continuación envía un mensaje **DHCPDISCOVER** a la dirección de broadcast para que sea recibido por todos los servidores DHCP del mismo segmento de red.

Mensaje **DHCPOFFERT**

Después de recibir el mensaje anterior, el servidor envía un mensaje **DHCPOFFER** presentando al cliente una dirección IP y otros parámetros de configuración de acuerdo con la definición de su fichero de configuración.

Mensaje **DHCPREQUEST**

En este mensaje el cliente DHCP solicita información de la dirección IP y los otros parámetros de configuración que el servidor proporcionó en el mensaje anterior; la dirección destino de este mensaje es la dirección de broadcast.

Mensaje **DHCPACK**

Después de recibir el mensaje **DHCPREQUEST** el servidor comprueba que la dirección solicitada está todavía disponible y que los parámetros de configuración son correctos, registra la dirección asignada y envía un mensaje de confirmación al cliente.

A partir de este punto del capítulo se van a presentar los resultados obtenidos durante el periodo de monitorización de tráfico DHCP (22 días) en la máquina **grunt.it.uc3m.es**.

El resultado de la captura es el siguiente:

```

1082740556.016234 0:c0:f0:3d:45:62 ff:ff:ff:ff:ff:ff
0.0.0.0.68 > 255.255.255.255.67:
      4500 0148 0300 0000 8011 36a6 0000 0000
      ffff ffff 0044 0043 0134 b549 0101 0600
      8105 8105 0e00 0000 0000 0000 0000 0000
      0000 0000 0000 0000 00c0 f03d 4562 0000
      0000 0000 0000 0000 0000 0000 0000 0000
      0000 0000 0000 0000 0000 0000 0000 0000
      0000 0000 0000 0000 0000 0000 0000 0000
      0000 0000 0000 0000 0000 0000 0000 0000
      0000 0000 0000 0000 0000 0000 0000 0000
      0000 0000 0000 0000 0000 0000 0000 0000
      0000 0000 0000 0000 0000 0000 0000 0000
      0000 0000 0000 0000 0000 0000 0000 0000
      0000 0000 0000 0000 0000 0000 0000 0000
      0000 0000 0000 0000 0000 0000 0000 0000
      0000 0000 0000 0000 6382 5363 3501 033d
      0701 00c0 f03d 4562 3204 a375 8bf8 3604
      a375 8b78 0c0c 4f45 4d43 6f6d 7075 7465
      7200 3708 0103 060f 2c2e 2f39 ff00 0000
      0000 0000 0000 0000

```

Para que el formato de la traza sea más legible y el análisis posterior de la información se simplifique, se aplica el filtro **dhcpcdump** sobre el fichero que contiene la captura.

```

TIME: 1082740556.016234
IP: 0.0.0.0.68 (0:c0:f0:3d:45:62) > 255.255.255.255.67
OP: 1 (BOOTPREQUEST)
HTYPE: 1 (Ethernet)
HLEN: 6
HOPS: 0
XID: 81058105
SECS: 3584
FLAGS: 0
CIADDR: 0.0.0.0
YIADDR: 0.0.0.0
SIADDR: 0.0.0.0
GIADDR: 0.0.0.0
CHADDR: 00:c0:f0:3d:45:62:00:00:00:00:00:00:00:00:00:00
SNAME: .
FNAME: .
OPTION: 53 ( 1) DHCP message type      3 (DHCPREQUEST)
OPTION: 61 ( 7) Client-identifier      01:00:c0:f0:3d:45:62
OPTION: 50 ( 4) Request IP address     163.117.139.248
OPTION: 54 ( 4) Server identifier      163.117.139.120
OPTION: 12 (12) Host name              OEMComputer
OPTION: 55 ( 8) Parameter Request List 1 (Subnet mask)
                                           3 (Routers)
                                           6 (DNS server)
                                           15 (Domainname)
                                           44 (NetBIOS name server)
                                           46 (NetBIOS node type)
                                           47 (NetBIOS scope)
                                           57 (Maximum DHCP message size)

```

El siguiente filtro se aplicará sobre un fichero que contenga únicamente los paquetes **request**, **offer**, **decline** enviados o recibidos por una máquina.

```
{
if (NR==1)
    time_ref=$1

else{

resultado=$1-time_ref
i++
array[i]=resultado
print array[i]
        total=total+array[i];
        time_ref=$1
    }

}
END {
total=total/NR
print("tiempo medio entre llegadas")
print(total)

for(k=1;k <=i ;k++){
var_interarrival=array[k]-total
    var_interarrival=var_interarrival*var_interarrival
    dur=dur+var_interarrival
}
print("varianza del tiempo entre llegadas")
dur= dur/NR

print(dur)

}
```

Este otro filtro sirve para averiguar el número de paquetes de cada tipo que se han transmitido durante el periodo de monitorización. Este tiempo supera los 20 días debido al limitado uso de este servicio.

```
{
if ($1== "TIME:"){

#solo se incrementa aqui el indice i
i++
    vector_time[i]=$2
}

#TIPO MENSAJE#

if ($1=="OPTION:" && $2==53){

    if($8==1)
        print("discover")
    if($8==2)
        print("offer")
    if($8==3)
        print("request")
    if($8==4)
        print("decline")
    if($8==5)
        print("ack")
    if($8==6)
        print("nack")
    if($8==8)
        print("release")
    if($8==8)
        print("inform")
}
}
```

| <i>Tipo Mensaje</i> | <i>Número paquetes</i> |
|---------------------|------------------------|
| Offer | 1852 |
| Discover | 5221 |
| Request | 2472 |
| Decline | 35 |
| Ack | 1862 |
| Nack | 216 |
| Release | 190 |
| Inform | 190 |
| Total | 12038 |

Tabla 8.1: Mensajes DHCP

A continuación se presentan unas figuras que corresponden a la función de distribución de probabilidad ($F(x)$) y la función de distribución de probabilidad complementaria ($F_c(x)$) del tiempo transcurrido entre la llegada de dos mensajes request, offer o decline. Se comprueba que estas variables se distribuyen según una función hiperbólica, no exponencial.

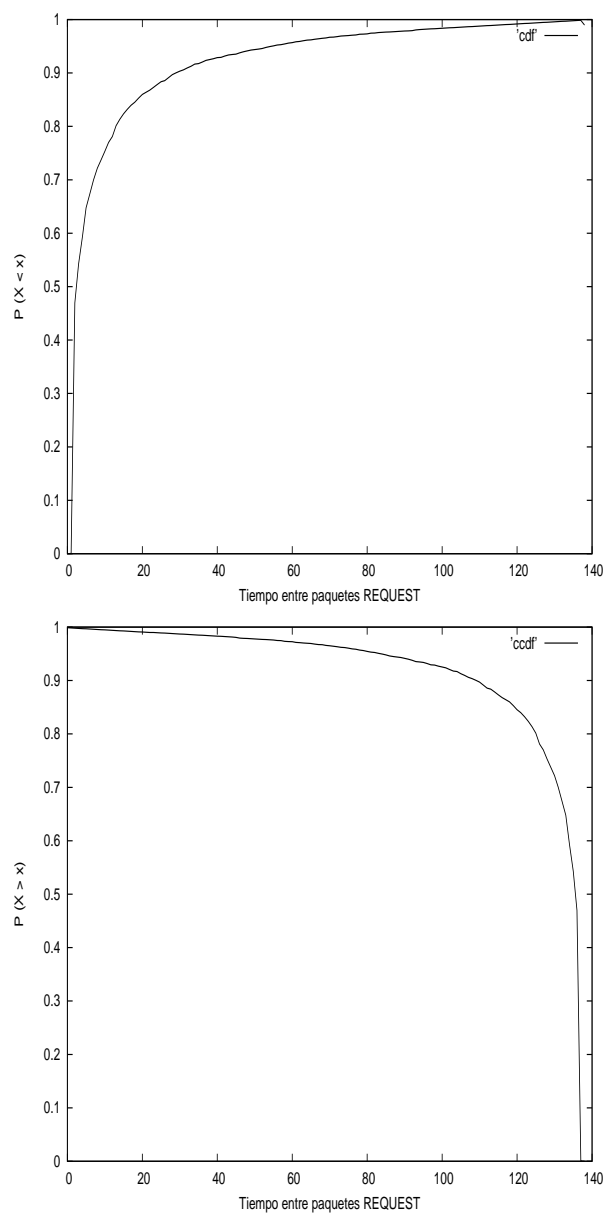


Figura 8.2: cdf (*cumulative distribution function*) y ccdf (*complementary cumulative distribution function*) del tiempo transcurrido entre mensajes *Request*

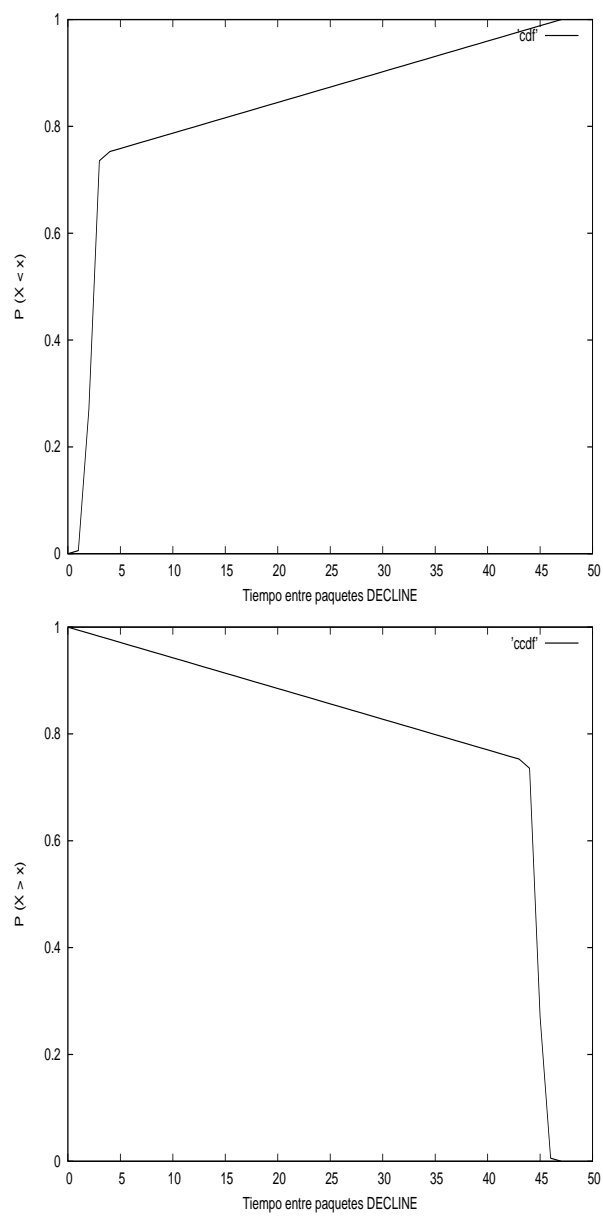


Figura 8.3: cdf (*cumulative distribution function*) y ccdf (*complementary cumulative distribution function*) del tiempo transcurrido entre mensajes *Decline*

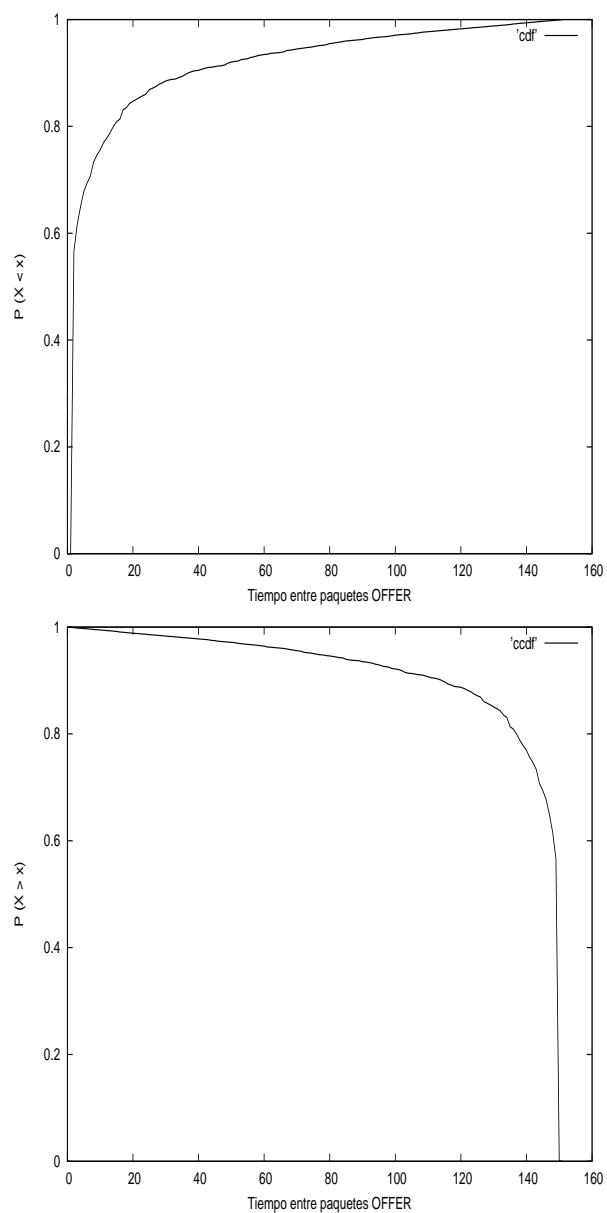


Figura 8.4: cdf (*cumulative distribution function*) y ccdf (*complementary cumulative distribution function*) del tiempo transcurrido entre mensajes *Offer*

Conclusiones y líneas futuras

9.1. Conclusiones

Comprender la naturaleza del tráfico en la subred del departamento de Ingeniería Telemática es esencial para las operaciones y la evaluación de rendimiento de la misma.

Los principales resultados de los análisis estadísticos de las medidas realizadas sobre el tráfico Ethernet son:

1. El tráfico es estadísticamente auto-similar (el valor del parámetro H es 0.6 - ligeramente superior al extremo inferior del intervalo $[0,5 - 1]$), independientemente de cuando hayan sido coleccionados los datos y en qué parte de la red se haya realizado la captura.
2. El grado de auto-similitud, medido en términos del parámetro H , es típicamente una función de la utilización de la red y se puede utilizar para medir la variabilidad (*burstiness*) del tráfico (a medida que aumenta su carácter intermitente, aumenta el valor de este parámetro).
3. Los principales componentes del tráfico Ethernet comparten las mismas características auto-similares del tráfico conjunto.

Las implicaciones de la naturaleza auto-similar del tráfico en las operaciones y evaluación de su rendimiento son:

1. Los modelos de fuente para usuarios Ethernet individuales exhiben variabilidad extrema en términos del tiempo entre llegada de paquetes (**síndrome de varianza infinita**).
2. Las medidas comúnmente usadas para la variabilidad, tales como el índice de dispersión (de cuentas), *peak-to-mean-ratio*, o el coeficiente de variación (para el tiempo entre llegadas), no son trascendentes para tráfico auto-similar sino que pueden ser remplazadas por el parámetro de Hurst.

3. La naturaleza de la congestión producida por los modelos de tráfico auto-similar difiere drásticamente de la observada en los modelos tradicionales.

9.2. Líneas futuras

Para el propósito de modelado de la dinámica del tráfico de la red de Ingeniería Telemática, FGN tiene la ventaja de proveer una descripción completa del proceso de tasa de tráfico resultante; ésto es, especificar su media, varianza y parámetro H es suficiente para caracterizar completamente el tráfico.

Dada esta ventaja sobre otras - típicamente incompletas - descripciones de la dinámica del tráfico de red, es importante conocer bajo qué condiciones el proceso FGN resulta adecuado y preciso.

[A. Erramilli et al., 1996] establece que el modelo FGN es apropiado porque:

- El tráfico es agregado sobre un gran número de conexiones independientes (garantizando gaussianidad por la expresión 4.1).
- Los efectos del control de flujo sobre cualquier conexión no se consideran (las escalas de tiempo consideradas son suficientemente grandes y se mantiene 4.8).
- El rango de escalas de interés para el problema de rendimiento coincide con la región de escala (donde se mantiene la relación 4.8).

En la práctica, estas condiciones se satisfacen en los backbone (altos niveles de agregación) y para escalas de tiempo mayores que el RTT (*round trip time*) de un paquete en la red.

9.2.1. Hacia modelos de carga de trabajo auto-similares no gaussianos

Una de las condiciones que justifican el empleo del proceso FGN como una descripción adecuada y precisa del tráfico de Ingeniería Telemática, es que el flujo de datos se constituye a partir de un gran número de conexiones independientes. La variabilidad de la tasa de tráfico en cada una de ellas puede no ser demasiado alta ($X_i^{(m)}$ se distribuye según una variable aleatoria con varianza finita).

Esta condición no es aplicable en la red de Ingeniería Telemática (tecnología de 100 Mbps), ya que, a pesar de la presencia de TCP, la tasa de tráfico de las conexiones individuales varía sobre varios órdenes de magnitud (de Kbps a Mbps), dependiendo de las condiciones de red. Así pues, para la comprensión del tráfico de la red bajo estudio, son de enorme relevancia los procesos que presentan colas pesadas en el tiempo y en el espacio (distribución estadística

heavy-tailed con varianza infinita para la duración y de la tasa de emisión de las conexiones individuales).

Para ilustrar lo que se ha mencionado anteriormente, se considera una conexión con periodos *on/off*, $X_i^{(m)}$, donde:

- La duración de los periodos *on/off* se ajusta a una función de distribución *heavy-tailed* con varianza infinita.
- La tasa de emisión de paquetes durante los periodos de actividad se distribuye según una variable aleatoria *heavy-tailed* con varianza infinita.

Centrando la atención en esta modificación del modelo *renewal reward process* [B.B.Mandelbrot, 1996] y [A. Erramilli et al., 1996] han demostrado recientemente que, cuando se estudia el proceso de tasa de tráfico conjunto $X^{(m)}$ definido en la ecuación 4.1, se obtiene un proceso estacionario con una distribución marginal estable de varianza infinita y un parámetro H dado por:

$$H = \frac{\beta - \alpha + 1}{\beta} \quad (9.1)$$

β denota el parámetro de la distribución *heavy-tailed* que ajusta la tasa de tráfico de las conexiones individuales.

α es el índice asociado a la distribución de la duración de los periodos *on/off*.

Ambos índices se asumen idénticos por simplicidad.

Si $\beta = 2$, la relación 9.1 se reduce a $H = (3 - \alpha)/2 \in (\frac{1}{2}, 1)$, valor que coincide con el valor de H en un proceso FGN (*Fractional Gaussian Noise*). Sin embargo, el proceso de superposición obtenido bajo la suposición de colas pesadas con varianza infinita de las duraciones y las tasas, no es gaussiano sino que tiene marginales *heavy-tailed*, implicando un aumento de la probabilidad de que la tasa de tráfico conjunto sea mucho mayor que el valor medio y que pueda tomar valores extremos (un fenómeno conocido como **intermitencia**).

En los procesos no gaussianos los parámetros estadísticos α (que especifican las marginales) y H no proveen una definición; existe un número de procesos dependientes, con incrementos estacionarios y con marginales estables con el mismo parámetro α y el mismo parámetro de auto-similitud. Sin embargo, los procesos FGN quedan completamente definidos a partir de las características estadísticas de segundo orden (varianza y parámetro Hurst), debido a la gaussianidad.

9.2.2. El comportamiento del tráfico de red en pequeñas escalas de tiempo

El análisis del tráfico medido en la red del departamento de Ingeniería Telemática y la comprensión de la estructura subyacente llevan a la conclusión de que este tráfico de área local es consistente con la auto-similitud asintótica o el comportamiento de escala a largo plazo, sus características en escalas de tiempo más pequeñas son muy diferentes de las observadas en escalas de tiempo mayores. Por tanto, para proveer una descripción más adecuada y completa del tráfico observado, sería necesario analizar sus características en escalas de tiempo pequeñas y comprender sus causas y efectos, pero ésto queda fuera del alcance de este trabajo.

Apéndice A

Selfis

Herramienta para el análisis de auto-similitud y dependencia a largo plazo A pesar del amplio uso de la auto-similitud y la dependencia a largo plazo del tráfico de red, existe confusión en torno a la identificación de tal fenómeno en el tráfico de una red real.

1. El exponente Hurst no puede ser calculado de una forma definitiva, sólo puede ser estimado.
2. Existen varios métodos diferentes para estimar el exponente de Hurst pero a menudo producen estimas diferentes.
3. No está claro cuál de los estimadores provee la estimación más precisa.

SELFIS es una herramienta software basada en java, que automatiza el análisis de auto-similitud.

Las principales características de *SELFIS* son las siguientes:

- Integra tres clases de funciones:
 - análisis de auto-similitud y dependencia a largo plazo.
 - Transformadas de Fourier.
 - Algoritmos de procesamiento de datos.
- Es implementada como una herramienta de software independiente, no es necesario software comercial adicional para utilizarlo.
- El diseño modular permite a otros investigadores contribuir con su propio código fuente.

SELFIS evita el empleo de sofisticados algoritmos en el análisis de datos reales. *SELFIS* es una herramienta software, basada en java, portable, expansible, orientada a objetos, libremente distribuida como un servicio a la comunidad de investigadores. Es una colección de metodologías para la estimación de auto-similitud y dependencia a largo plazo y algoritmos de procesamiento de series temporales. También incorpora varios estimadores

de dependencia a largo plazo que revelan características diferentes de las series analizadas. Asimismo ofrece metodologías de procesamiento de datos y transformaciones tales como wavelets, transformada de Fourier, test de estacionariedad. Además, *SELFIS* provee la posibilidad de sintetizar secuencias dependientes a largo plazo, para la que incluye generadores de muestras FGN.

Una de las desventajas de este programa es el retardo en cada una de las acciones, problema agravado por el tamaño de los ficheros de entrada (58 Mbytes). Es por ello, por lo que el proceso de estimación del parámetro H del tráfico de la red de Ingeniería Telemática se ha llevado a cabo utilizando tres programas codificados en C (disponibles en <ftp://ftp.ics.uci.edu/pub/duke/self-similar/>), que implementan los algoritmos descritos en el capítulo 3.

Analizadores de protocolos

Para analizar el comportamiento de los protocolos de red es preciso disponer de una herramienta capaz de monitorizar el tráfico existente en una interfaz y mostrarlo de una forma legible. Esta herramienta se denomina **analizador de protocolos de red** y registra parte o el contenido completo de cada paquete.

El post-procesado de esta información se denomina *retrospective analysis*.

Para capturar todo el tráfico, los analizadores de protocolos colocan la tarjeta de red en modo promiscuo, para ver todo el tráfico generado por todos los equipos que comparten el mismo conjunto de cables y concentradores (hubs).

La arquitectura software para un protocolo de red en una máquina Linux con tarjeta Ethernet se describe de esta forma:

El analizador de protocolos corre como una aplicación, comunicándose con un componente del kernel¹.

El socket de filtro (*Linux Socket Filter*) actúa como intermediario entre el analizador de protocolos y el controlador de la tarjeta de red Ethernet (*Ethernet device driver*); coloca la tarjeta de red en modo promiscuo y obtiene una copia de todo el tráfico entrante (desde la red) y saliente (hacia la red). El socket de filtro procesa este tráfico y lo transfiere al analizador de protocolos, que lo presenta al usuario.

Existen muchas herramientas útiles dependiendo del S.O., tipo de red, ... Una de estas herramientas es un sniffer de red, basada en la librería de captura de paquetes (libpcap) y que además funciona en plataformas tanto Win32 como GNU/Linux-UNIX. Esta herramienta es **TCPDump**, que funciona en línea de comando. Otro analizador de protocolos es **ethereal**, que es una aplicación gráfica.

¹El kernel de un sistema operativo es la parte central o núcleo del sistema

B.1. Ethereal

Esta herramienta se utilizó en el comienzo del trabajo para coleccionar trazas de tráfico en la red de Ingeniería Telemática. Tiempo después fue reemplazada por el analizador de protocolos *TCPDump* debido a la simplicidad y facilidad de uso de este último.

Ethereal es un potente analizador libre de protocolos de redes, para máquinas Unix y Windows. Permite capturar los datos directamente de una red o obtener la información a partir de una captura en disco (puede leer más de 20 tipos de formato distintos). Destaca también por su impresionante soporte de más de 300 protocolos, gracias sin duda a la licencia GPL y sus más de 200 colaboradores de todo el mundo. En la ventana principal se reconocen dos áreas de despliegue:

1. Resumen de paquetes capturados, un paquete por línea.
2. Detalles de encabezado de protocolos para el paquete seleccionado.

La ventana de estado muestra en tiempo real la cantidad de paquetes capturados, en total y de algunos tipos corrientes.

La situación de captura se mantiene hasta que se presiona el botón *stop*. Después de unos instantes aparecen los paquetes capturados, tal cual se ve en la imagen de la ventana principal; si se activa la opción de actualizar lista de paquetes en tiempo real (*Update list of packets in real time*) estos se visualizan a medida que son capturados

B.2. TCPDump

TCPDump (en Windows, WINDump) es un programa cuya utilidad principal es analizar el tráfico que circula por la red. Se apoya en la librería de captura **pcap**, que presenta una interfaz uniforme y que esconde las peculiaridades de cada sistema operativo a la hora de capturar tramas de red. Para seguir el manual es necesario unos conocimientos básicos del protocolo TCP/IP, [Stevens et al., 1995].

Aunque viene incluido en la mayoría de las distribuciones de Linux, sus fuentes pueden encontrarse en <http://www.tcpdump.org/>.

En primer lugar hay que definir la interfaz de red en la que debe escuchar el analizador de protocolos; cuando se ejecuta sin parámetros elige la interfaz *eth0* (en una máquina UNIX), mientras que en Windows hay que indicar explícitamente la interfaz de red a monitorizar.

Para averiguar las interfaces en cualquier máquina UNIX se ejecuta el comando `ifconfig -a` Unix, que imprime por pantalla las interfaces de la máquina, así como sus parámetros de configuración.

```
lm011:~ /sbin/ifconfig -a
```

```
eth0      Link encap:Ethernet  HWaddr addr
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:646760 errors:0 dropped:0 overruns:0 frame:0
          TX packets:449673 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:100
          Interrupt:5 Base address:0x2c20

eth1      Link encap:Ethernet  HWaddr addr
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:1321583 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1778135 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:100
          Interrupt:9 Base address:0x3000

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          UP LOOPBACK RUNNING  MTU:3924  Metric:1
          RX packets:39747 errors:0 dropped:0 overruns:0 frame:0
          TX packets:39747 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
```

(en este ejemplo están borradas las direcciones IP)

Cuando se "lee" la red, puede que no interese que TCPDump intente resolver los nombres de las maquinas (pueden que no estar dadas de alta en el DNS, por motivos de seguridad, ...). Para ello se dispone de la opción **-n**.

Para establecer la longitud de los datos que captura TCPDump se emplea la opción **-s len**, donde *len* es la longitud que interesa. Por defecto TCPDump sólo captura los primeros 68 bytes, que resulta útil lo único que se quiere son las cabaceras IP, TCP o UDP, pero en otros casos truncan los datos.

Se puede ajustar la longitud de la captura a la MTU del medio con esta opción. Por ejemplo para capturar toda la trama ethernet se puede usar **-s 1500**.

En función de la cantidad de información que TCPDump debe interpretar, se puede utilizar **-v**, **-vv**, **-vvv**, aumentando el grado de información con cada una de las opciones.

Si se quiere imprimir el contenido del paquete se puede utilizar la opción **x** (para imprimir en ASCII se utiliza la opción **-X**). La longitud que

imprime viene determinada por la opción `-s` o los 68 bytes que usa captura por defecto.

Se puede trabajar *offline* con TCPDump. Si se quiere grabar la captura para posteriormente leerla y analizarla se utiliza la opción `-w file` donde *file* es el nombre del fichero donde quedará almacenada la captura de datos. Posteriormente se puede leer y analizar *offline* con la opción `-r file` ².

B.2.1. Interpretando la salida

Lo primero que hay que decir es que la salida depende del protocolo que estemos analizando. Para empezar comentar que todas las capturas del tcpdump tienen como primer campo una marca de tiempo, que indica cuando ha sido capturado el paquete.

Protocolo ARP/RARP

El protocolo ARP (*Address resolution protocol*) sobre Ethernet está documentado en la RFC 826 <http://www.faqs.org/rfcs/rfc826.html>. RARP se puede encontrar en la RFC 1293 <http://www.cis.ohio-state.edu/cgi-bin/rfc/rfc1293.html>. Estas peticiones aparecen de la siguiente manera:

```
18:33:49.908612 arp who-has 192.168.1.2 tell 192.168.1.1
18:33:49.908691 arp reply 192.168.1.2 is-at 0:2:a5:ee:ec:10
```

En este caso, la máquina 192.168.1.1 pregunta por la dirección ethernet 192.168.1.2 (se supone que ambas máquinas están ubicadas en el mismo segmento de red).

²Este fichero también puede ser procesado por otros analizadores, tales como Ethereal (<http://www.ethereal.com>).

Protocolo TCP

El protocolo TCP se define en la RFC 793 <http://www.faqs.org/rfcs/rfc794.html>. La linea general de un paquete TCP es:

```
src > dst: flags [dataseq ack window urgent options]
```

Los campos **src**, **dst** y **flags** estan siempre presentes. El significado de dichos parámetros es:

Src: Dirección y puerto origen. En caso de no especificar el parámetro -n se intenta resolver el nombre via DNS y el se busca el nombre del puerto vía (normalmente en los Unix en /etc/services).

Dst: Dirección y puerto destino, exactamente igual que el caso anterior.

Flags: Flags de la cabecera TCP. Puede ser " ." (no hay flags, o una combinación de S (SYN), F (FIN), P (PUSH), W (reducción de la ventana de congestión), E (ECN eco).

Dataseq: Número de secuencia del primer byte de datos en este segmento TCP. El formato es "primero:último(n)", que significa que desde el primero al último (sin incluir) hay un total de n bytes de datos ³.

Ack: Número de asentimiento, siguiente número de secuencia que se espera recibir ⁴.

Win: Tamaño de la ventana de recepción.

Options: Indica la existencia de opciones. Si hay, van entre < y >.

Protocolo UDP

Un paquete UDP se imprime de la siguiente manera:

```
origen.srcport > destino.dsport: udp len
```

Origen: Nombre o dirección origen.

Src-port: Puerto origen.

Destino: Nombre o dirección destino.

Dst-port: Puerto destino

Len: Longitud de los datos de usuario.

En algunos casos, puede interpretar protocolos que vayan encapsulado en los paquetes UDP (NFS o DNS).

³Los segmentos SYN también ocupan un número del espacio de secuencia.

⁴Para los segmentos SYN también hay asentimiento.

Fragmentos de datagramas IP

Los datagramas fragmentados se indican con una expresión al lado de los mismos entre paréntesis:

`(frag id:size@offset+) (frag id:size@offset)`

Id: identificador del fragmento.

Size: tamaño del fragmento.

Offset: posición del fragmento en el datagrama original. El significado de "+" al final del campo es la existencia o no de más fragmentos.

Apéndice C

Tcp-Reduce

Esta herramienta es una colección de scripts que, a partir de un fichero (formato binario) con la captura completa, extraen la información de cada conexión TCP (establecida y finalizada durante el periodo de monitorización) y la presenta en pantalla.

- `tcp-reduce`.
- `tcp-conn`.
- `tcp-summary`.

Formato de los ficheros reducidos Cada línea ASCII del fichero reducido (obtenido tras ejecutar el script *tcp-reduce* sobre el fichero que contiene la captura tcpdump) contiene 8 o 9 columnas y resume una conexión TCP.

1. La primera columna indica el instante en el que comenzó la conexión (primer paquete SYN).
2. Duración de la traza en segundos o ? si la traza no muestra el fin de la conexión (no se han recogido paquetes con el flag FIN o RST activado).
3. Protocolo empleado en el nivel de aplicación. En general es derivado del número de puerto utilizado por el receptor, excepto en el caso de ftp-data. En este caso el paquete SYN se envía al puerto 20 de TCP. Los puertos no identificados se almacenan como *other-XXXX* si no son privilegiados (puerto menor que 1024) o *priv-XXXX* si el puerto es mayor de 1024.
4. Bytes enviados por el que inició la conexión o ? si no está disponible (la conexión no ha terminado o ha terminado con RST).
5. Bytes enviados por el receptor de la conexión, o ? si no está disponible.
6. Host local y remoto que participaron en la conexión.

El estado en el que acaba una conexión puede ser:

- SF normal completado SYN/FIN.
- REJ conexión rechazada.
- S0 (estado 0): no se responde a la solicitud de inicio de conexión.
- S1 (estado 1): conexión establecida.
- S2 (estado 2): la conexión se ha establecido correctamente y el host origen ha cerrado la conexión por su parte.
- S3 (estado 3): la conexión se ha establecido correctamente y el host destino ha cerrado la conexión por su parte.
- S4 (estado 4): en las trazas de tráfico no se ha registrado el mensaje de solicitud de conexión pero si el mensaje de confirmación de recepción.
- RSTOSn el host origen interrumpió la conexión (envió un paquete con el flag RST activado) en el estado n.
- RSTRSn el host origen interrumpió la conexión en el estado n.
- SS el paquete de solicitud de conexión se registró cuando la conexión estaba parcialmente cerrada.
- SH el cierre de una conexión en estado 0 se produjo antes de registrar el paquete de confirmación de la solicitud de conexión (flag SYN activado).
- SHR el cierre de una conexión en estado 4 se produjo antes de registrar el paquete de solicitud de conexión.
- OOS1 El paquete de confirmación no se corresponde con el paquete de solicitud de conexión.
- OOS2 El paquete de solicitud de conexión se retransmitió con un número de secuencia diferente.

La salida del script *tcp-summary* se presenta en seis columnas:

- Protocolo TCP.
- Número de conexiones intentadas en cada puerto (identificador del protocolo en el nivel de aplicación).
- Número de kilobytes (KB) transferidos en ambas direcciones (host origen-destino y destino-origen).
- Porcentaje de conexiones que finalizaron en estado SF.

- Porcentaje de conexiones iniciadas por equipos del segmento de red del departamento de Ingeniería Telemática (hosts locales).

No se consideran aquellas conexiones en las que se transfiere una ínfima cantidad de bytes (el valor de este umbral es configurable).

Presupuesto del proyecto

El presupuesto de este proyecto, de duración 6 meses, consta de tres apartados: el cálculo del presupuesto de ejecución material, el coste de la dirección de obra y, por último, el presupuesto total, suma de los dos conceptos anteriores ¹.

D.1. Presupuesto de ejecución material

Se incluyen en este presupuesto los gastos en herramientas empleadas, tanto hardware como software, así como la mano de obra.

D.1.1. Gastos de material

| Concepto | Precio | Uso (meses) | Amortización (años) | Total |
|------------------------------|--------|-------------|---------------------|-------|
| Ordenador i686 | 590 | 6 | 1 | 295 |
| Impresora Láser | 650 | 1 | 5 | 130 |
| Material de oficina | 150 | | | 150 |
| Microsoft Office 2003 | 300 | 6 | 3 | 50 |
| Sistema Operativo Windows XP | 165 | 12 | 3 | 55 |
| | | | | |
| TOTAL | | | | 680 |

Tabla D.1: Gastos de material

El resto de herramientas software empleadas no se contabilizan porque son de libre distribución.

¹Todas las cantidades que aparecen están contempladas en Euros.

D.2. Coste de mano de obra

| Concepto | Sueldo mensual | Meses | Total |
|----------------------|----------------|-------|-------|
| Ingeniero Telemático | 1200 | 4 | 4800 |
| Mecanógrafo | 600 | 2 | 1200 |
| | | | |
| TOTAL | | | 6000 |

Tabla D.2: Costes mano de obra

| |
|--|
| TOTAL PRESUPUESTO DE EJECUCIÓN MATERIAL 6680 Euros |
|--|

D.2.1. Dirección del proyecto

La dirección del proyecto es llevada a cabo por un ingeniero que realiza las funciones de dirección, administración y análisis de requisitos. Sus honorarios ascienden a 12.000 Euros.

D.2.2. Importe total del proyecto

| | |
|-----------------------------------|--------------|
| Presupuesto de ejecución material | 6680 |
| Dirección del proyecto | 12000 |
| | |
| TOTAL | 18680 |

Tabla D.3: Coste total del proyecto

El importe total del proyecto asciende a DIECIOCHO MIL SEISCIENTOS OCHENTA EUROS.

El ingeniero,

Fdo: Patricia Bravo García
Madrid, Junio de 2004

Bibliografía

- [B.B.Mandelbrot,1996] B.B.Mandelbrot, 1996 The fractal Geometry of Nature, Nueva York: Freeman and Company,1997. In *ACM SIGCOMM 93*, San Francisco, CA.
- [Babic et al., 1998] Babic, G., Vandalore, B., and Jain, R. (1998). Analysis and modeling of traffic in modern data communication networks. Technical Report OSU-CISRC-1/98-TR02, Dep. of Computer and Information Science, Ohio State University.
- [Frost and Melamed, 1994] Frost, V. S. and Melamed, B. (1994). Traffic modeling for telecommunications networks. *IEEE Communications Magazine*, pages 70–81.
- [W. Leland, 1990] W. Leland y W. Wilson, 1991 High time-resolution measurement and analysis of LAN traffic: Implications for LAN interconnection
- [Paxson and Floyd, 1995] Paxson, V. and Floyd, S. (1995). Wide-area traffic: The failure of poisson modeling. *IEEE/ACM Transactions on Networking*, 3(3):226–244.
- [Leland et al., 1993] W. Leland, 1993 On the self-similar nature of ethernet traffic. In *ACM SIGCOMM 93*, San Francisco, CA.
- [Leland et al., 1994] Leland, W., Taqqu, M., Willinger, W., and Wilson, D. (1994). On the self-similar nature of ethernet traffic (extended version). *IEEE/ACM Transactions on Networking*.
- bibitem[Aracil, 1996]ARA96b Aracil, J. (1996). Características del tráfico en la internet e implicaciones para el análisis y dimensionamiento de redes de ordenadores.
- [Grossglauser and Bolot, 1996] Grossglauser, M. and Bolot, J.-C. (1996). On the relevance of long-range dependence in network traffic. Technical Report 2830, INRIA, Sophia-Antipolis (France).
- [Ng, 1997] Ng, C. H. (1997). *Queueing Modelling Fundamentals*. John Wiley and Sons.

- [Kleinrock, 1974] Kleinrock, L. (1974). *Queueing Systems. Volume I: Theory*. John Wiley & Sons.
- [M.E. Crovella, 1996] M.E. Crovella y Bestavros, 1996 "Self-Similarity in World Wide Web. Evidence and Possible Causes"
- [A. Erramilli et al., 1996] A. Erramilli, 1996 Experimental queueing analysis with long-range dependent packet traffic; 4:209-223
- [A. Erramilli et al., 1996] A. Erramilli, 1996 Experimental queueing analysis with long-range dependent packet traffic; 4:209-223 In *ACM SIGCOMM 93*, San Francisco, CA.
- [Fowler and Leland, 1991] Fowler, H. J. and Leland, W. E. (1991). Local area network traffic characteristics, with implications for broadband network congestion management. *IEEE Journal on Selected Areas in Communications*, 9:1139–1149.
- [B.B.Mandelbrot, 1996] B.B.Mandelbrot, 1996 Long-run linearity, locally Gaussian processes, H-spectra and infinite variances In *ACM SIGCOMM 93*, San Francisco, CA.
- [Pappu, 1996] Pappu, S. K. (1996). *Performance Analysis of Input and Output Queueing Strategies of an ATM Switch under Self-similar Traffic*. PhD thesis, Clemson University.
- [Heyman and Lakshman, 1996] Heyman, D. P. and Lakshman, T. (1996). Source models for VBR broadcast-video traffic. *IEEE/ACM Transactions on Networking*, 4(1):40–48.
- [Tsybakov, 1997] Tsybakov, B. (1997). Overflow probability in an ATM queue with self-similar input traffic. In *IEEE International Communications Conference ICC '97*, Montreal.
- [Jain and Routhier, 1986] Jain, R. and Routhier, S. A. (1986). Packet trains—measurements and new model for computer network traffic. *IEEE Journal on Selected Areas in Communications*, SAC-4(6):986–995.
- [Raj Jain, 1986] Raj Jain, 1986 Packet trains-measurements and a new model for computer network traffic
- [Karol et al., 1987] Karol, M. J., Hluchyj, M. G., and Morgan, S. P. (1987). Input versus output queueing on a space-division packet switch. *IEEE Transactions on Communications*, 35(12):1347–1356.
- [Willinger et al., 1996] Willinger, W., Taqqu, M. S., and Erramilli, A. (1996). A bibliographical guide to self-similar traffic and performance modeling for modern high-speed networks. In Kelly, F., Zachary, S., and

- Ziedins, I., editors, *Stochastic Networks: Theory and Applications*, pages 339–366, Oxford. Clarendon Press.
- [Kokoszka and Taqqu, 1996] Kokoszka, P. S. and Taqqu, M. S. (1996). Parameter estimation for infinite variance fractional ARIMA. *The Annals of Statistics*, 24:1880–1993.
- [Lau et al., 1995] Lau, W.-C., Erramilli, A., Wang, J. L., and Willinger, W. (1995). Self-similar traffic generation: The random midpoint displacement algorithm and its properties. In *IEEE ICC'95*, Seattle.
- [Paxson, 1995] Paxson, V. (1995). Fast approximation of self-similar network traffic. Technical Report LBL-36750, Lawrence Berkeley Laboratory, EECS Division, University of California, Berkeley.
- [Norros, 1995] Norros, I. (1995). On the use of fractional brownian motion in the theory of connectionless networks. *IEEE Journal on Selected Areas in Communications*, 13(6):953–962.
- [Norros et al., 1999] Norros, I., Mannersalo, P., and Wang, J. L. (1999). Simulation of fractional brownian motion with conditionalized random midpoint displacement. Technical report, COST 257, to appear in *Advances in Performance Analysis*.
- [Robert and Le Boudec, 1995a] Robert, S. and Le Boudec, J.-Y. (1995a). Can self-similar traffic be modeled by markovian processes? Technical Report COST 242 TD95-26, EPFL - DI - LRC, CH-1015 Lausanne, Switzerland.
- [Robert and Le Boudec, 1995b] Robert, S. and Le Boudec, J.-Y. (1995b). A markov modulated process for self-similar traffic. Technical report, Laboratoire de Reseaux de Communication (LRC), Lausanne.
- [Stevens et al., 1995] Stevens, R. and Addison (2000) *TCP/IP Illustrated*, Volumen 1.